The Dissertation Committee for Paolo Gatto
certifies that this is the approved version of the following dissertation:

Modeling bone conduction of sound in the human head using *hp*-finite elements

Committee:

Leszek F. Demkowicz, Supervisor

Todd Arbogast

Mark F. Hamilton

J. Tinsley Oden

Jorge Peters

Robert A. van de Geijn

Modeling bone conduction of sound in the human head using *hp*-finite elements

by

Paolo Gatto, Laurea Specialistica

DISSERTATION
Presented to the Faculty of the Graduate School of
The University of Texas at Austin
in Partial Fulfillment
of the Requirements
for the Degree of

DOCTOR OF PHILOSOPHY

THE UNIVERSITY OF TEXAS AT AUSTIN
December 2012

To mom and dad.

Acknowledgments

This work as been possible through the collective effort of a number of individuals with outstanding professional qualities. In first place, I would like to thank Drs. E. and M. Bleszyński for initiating the project that made this dissertation possible. I am also indebted to Dr. W. Rachowicz, who helped me with code verification in the early stages of this project, when I was a novice to the Fortran language, and later provided routines that I heavily borrowed from to implement an error indicator. The first numerical simulations were run by Dr. M. Paszyński, who developed the interface to the MUMPS parallel solver.

In this work I was faced with a substantial coding component. Software development would have not been possible without the help of other fellow graduate students, specially Kyungjoo Kim, who wrote most of the refinement package and an interface to his excellent solver, UHM; and Jeffrey Zitelli who provided feed back as a user.

I would also like to thank Dr. J. Kurtz for developing an interface with the Visualization Tool Kit, and Dr. M. Hamilton for the many productive discussions about the numerical results that I obtained.

A special thank goes to all the friends from ICES that have made my life in Austin an exciting and rewarding experience under a human point of view. All of you will be dearly remembered.

Last but not least, I would like to thank professor L. Demkowicz, for his support, dedication, and advice. His professional guidance, and hard-working ethics have provided me with a unique graduate school experience.

Modeling bone conduction of sound in the human head using *hp*-finite elements

Publication No. ____________________

Paolo Gatto, Ph.D.
The University of Texas at Austin, 2012

Supervisor: Leszek F. Demkowicz

Hearing impairment is the primary disability among military personnel. Proximity to aircraft engines or impulse noise from large caliber weapons expose soldiers to sound pressure levels that may easily exceed the pain threshold value of 120 dB. While ear plugs protects against the air-borne sound waves, they do not shield from the bone-conducted sound. This work proposes a numerical model for investigating the bone-conducted sound in the human head, and quantifying its physiological effects. The main difficulty of the problem arises from the lack of fundamental knowledge regarding the transmission of acoustic energy through non-airborne pathways to the cochlea, the receptor that converts the mechanical signal into electric impulses sent to the brain. I employed a fully coupled model based on the acoustic/elastic interaction problem with a detailed resolution of the cochlea region and its interface with the skull and the air pathways. The model was simulated via *hp*-finite element

approximation through a code, `hp3d`, that was developed inside Demkowicz's research group at The Institute for Computational Engineering and Sciences (ICES). The numerical results provide an insight into this fundamental, long standing research problem.

The fact that vibration of the skull causes a hearing sensation has been known since the 19th century and termed "hearing by bone conduction." Although there has been more than a century of research on the phenomenon, its physiology is not completely understood yet. For an early account of the bone conduction as cause of partial deafness see [82]. Further summaries of pioneering works can be found in Hood (Journal of the Acoustical Society of America, 34: 1325-1332, 1962) and Naunton (Chapter I in Modern Development in Audiology, J. Jerger, ed., Academic Press: New York, 1963). Lately, new insights into the physiology of hearing by bone conduction have been reported. Factors contributing to bone conduction of sound, such as inertia of the middle ear ossicles and of the cochlear fluids, and sound radiated into the external ear canal have been investigated by S. Stenfelt and R. Goode in [78]. Experimental data regarding the first simultaneous sound pressure measurements in *scala vestibuli* and *scala tympani* of the cochlea in temporal bones of human cadavers were presented by E. Olson et al. in [59].

As anticipated, the project has driven the development of a new *hp*-adaptive finite element code for multiphysics applications and coupled problems, see [22]. By expanding the technology previously developed inside Demkowicz's group, see [21, 25], I have contributed to build a code that supports variable order elements of all shapes (hexahedra, tetrahedra, prisms, and pyramids), both anisotropic (hexahedra, prisms, and pyramids) and isotropic (hexahedra, tetrahedra, and prisms) refinements, and enables computations on 1-irregular meshes, i.e., only one level of hanging nodes is allowed. While interfacing with an external geometry package is certainly possible, `hp3d` comes with its one geometry package (GMP) that constructs maps $\mathbf{x}_b$ for each block by combining transfinite or parametric transfinite interpolation for edges and faces of each block in a bottom-up fashion.

The head model I managed to build uses algebraic surfaces only (spheres, cylinders, cones, and planes) and was generated using NETGEN, a tetrahedral mesh generator developed by Joachim Schöberl, see [70]. Although rather primitive, the model matches a typical head in terms of size of the ear canal, thickness of the skull and the tympanic membrane, and dimensions of the basilar membrane. A first attempt to mesh thin-walled structures, namely membranes and spherical shells, with tetrahedral meshes led to a prohibitive number of elements and highly distorted tetrahedra. The problem was overcome by using prismatic elements in the membranes and shells, at the expense of introducing pyramids as connecting elements between the prisms and the tetrahedra.

The variational formulation I employed for this coupled elasticity/acoustics problem is classical and has been extensively studied over the last two decades, see, e.g., [39]. Such a formulation is obtained by performing a *weak coupling* between the acoustic and elastic variables, namely the involved variables (elastic displacement and acoustic pressure) are not forced to match each other on the elastic-acoustic interface. Instead, the variational formulation incorporates additional off-diagonal flux terms representing the interaction between the two fields and resulting from the interface conditions. As a result, the coupling terms represent compact perturbations of the terms on the main diagonal corresponding to elasticity and acoustics. This guarantees automatically that the Galerkin discretization of the problem will be *asymptotically stable.*

The problem is naturally posed in an unbounded domain. Since the finite elements method can handle only bounded domains, a truncation of the unbounded physical domain to a bounded computational domain is necessary. This calls for the development of an *ad hoc* boundary condition to be imposed on the outer boundary, such that the original problem is not essentially altered. Among various techniques such as the Infinite Element, Dirichlet-to-Neumann operators, and Absorbing Boundary Conditions, the Perfectly Matched Layer (PML) has achieved popularity due to its conceptual simplicity, ease of implementation and good performance. The original idea, due to Bèrenger [4] and subsequently developed by Chew and Weedon [17], is based on the analytic continuation of a real function into the complex plane.

The code supports variable order isoparametric and exact geometry elements. In the case of parametric elements, the exact geometry map $\mathbf{x}_b \in H^r$, for some $r \geq 1$, is replaced by a polynomial map $\mathbf{x}_{hp}$ and the convergence of $\mathbf{x}_{hp}$ to $\mathbf{x}_b$ in H^r is essential in order not to spoil the expected convergence rate for the solution of the problem of interest. A natural choice for building $\mathbf{x}_{hp}$ is Projection-Based Interpolation (PBI). As discussed in [25], PBI enjoys locality, global conformity, and optimality. In other words, the construction is done element-wise, returns an object that belongs to the proper energy space over the entire mesh, and guarantees optimal rates of convergence, both with respect to the element size h and the polynomial order

p.

The support of variable order elements allows us to set up a non uniform initial p-mesh. While the natural choice for the order of approximation is $p = 2$, we locally raise the order to $p = 4$ to prevent locking of membranes, i.e., plates, and to $p = 5$ to prevent locking of shells. Furthermore, the possibility of varying the order would accommodate the tiny elements caused by a domain with a potentially high curvature. In order to drive h-adaptivity I developed an explicit *a posteriori* error indicator for the elasticity/acoustics problem.

Calibration of material constants is essential when studying a bio-mechanical problem. In the case of interest, material parameters found in the literature can differ by orders of magnitude. This is due to the difficulty of conducting experiments to investigate such material properties, and to the objective fact that material properties vary within the same tissue or bone. This was mostly evident when searching for material constants for the skull and the brain. Nevertheless, it is perfectly legitimate to investigate the sensitivity of the results on the parameters of the model. To this aim, I have conducted and extensive parametric study, which investigates the response over the whole spectrum of audible frequencies (20 Hz–20 kHz), the effects of directionality of the incident plane wave, and blocking the ear canal with a plug. I believe that the results thus obtained contribute to a better understanding of bone conduction of sound.

Table of Contents

Acknowledgments **v**

Abstract **vii**

List of Tables **xiv**

List of Figures **xv**

Chapter 1. A Coupled Acoustics Elasticity Problem **1**

1.1 Acoustics 1
1.2 Elasticity 9
1.3 Coupling Acoustics and Elasticity 13
1.4 Mathematical Analysis 19
1.5 Perfectly Matched Layer 29

Chapter 2. Finite Element Code Design **48**

2.1 Geometry Modeling 48
2.2 Shape Functions 52
2.3 Exact Geometry Elements & Isoparametric Elements 55
2.4 Code Data Structure 60
2.5 Element to Nodes Connectivities 70
2.6 Mesh Refinements 77
2.7 Mesh Reconciliation & Mesh Regularity 84
2.8 Unassembled Hyper-Matrix Solver 87

Chapter 3. The Head Problem **91**

3.1 Problem Set-Up 91
3.2 Non-dimentionalization 98
3.3 Geometry Convergence Test 102

3.4 *A posteriori* Error Indicator 105
3.5 Element Routine Testing 117
  3.5.1 Model Problems 117
  3.5.2 Air Force problem 129
3.6 Parametric Study 132

Chapter 4. Conclusions and Future Work **163**

Appendices **169**

Appendix A. G^1-Interpolation Scheme for Unstructured Triangular Grids **170**
A.1 Scheme Construction 170
  A.1.1 Twist Compatibility Condition 170
  A.1.2 Curves Network Reconstruction 171
  A.1.3 G^1 Constraint 173
  A.1.4 Bernstein Polynomials and Bezier Patches 174
  A.1.5 Computation of Radial Derivative 176
  A.1.6 Radial Derivative Reconstruction 177
  A.1.7 Patch Reconstruction 180
A.2 Scheme Implementation 181
  A.2.1 Wireframe Reconstruction 181
  A.2.2 Patch Reconstruction 182
  A.2.3 Scheme Testing 185

Bibliography **190**

Vita **202**

List of Tables

2.1 Explanation of boundary condition flags. 68

3.1 Material constants. The wave length refers to $f = 2000\,\text{Hz}$. 94
3.2 Geometry convergence rates for the mesh shown in Figure 3.4. . . . 104
3.3 Geometry convergence rates for the mesh shown in Figure 3.5. . . . 106
3.4 Numerical results for I Problem. Uniform refinements were used. . . 119
3.5 Numerical results for I Problem. Adaptive refinements were used. . . 120
3.6 Numerical results for II Problem. Uniform refinements were used. . . 124
3.7 Numerical results for II Problem. Adaptive refinements were used. . 125
3.8 Numerical results for II Problem. Adaptive refinements and rescaled error norms were used. 126
3.9 Numerical results for III Problem. Uniform refinements were used. . 128
3.10 Numerical results for III Problem. Adaptive refinements were used. . 128
3.11 Numerical results for III Problem. Adaptive refinements and rescaled error norms were used. 129
3.12 Distribution of error indicator over incident directions. 130

List of Figures

1.1 PML layer and truncating domains. 41

2.1 Extrusion of a membrane inside a cylindrical shell. The red circle represents the bounding cylinder, while the black circle represents the conforming cylinder. 52
2.2 Hierarchical shape functions for elements of all shapes. Column-wise: vertex mode, even edge bubble, odd edge bubble, lowest order face bubble, higher order face bubble. 55
2.3 Master, reference, and physical space. 58
2.4 Node numbering and local orientations for 2D elements. 60
2.5 Node numbering and local orientations for tetrahedral element. 61
2.6 Node numbering and local orientations for prismatic element. 62
2.7 Node numbering and local orientations for pyramidal element. 63
2.8 Triangle orientations. 65
2.9 Quadrilateral orientations. 66
2.10 Some node refinement kinds, showing the enumeration and orientations of son nodes. The small numbers next to the vertices indicate the orientation of the father node. 72
2.11 Isotropic refinement of a triangular face. The local enumeration of the nodes of the first son element is indicated in blue. Some of the nodes of the father element are indicated in green. The black numbers are the local enumeration of the son-nodes, as dictated by the father node. 74
2.12 Discrepancy between son-element local orientations (indicated in blue) and orientations as inherited by the father node (indicated in black). 76
2.13 Supported prism, tetrahedron, and pyramid refinements. 78
2.14 Isotropic tetrahedron refinement. 79

3.1 Constructive Solid Geometry model built *via* Netgen and the Geometry Modeling Package. 139
3.2 Middle ear detail of the model. 140
3.3 Initial mesh order of approximation. 141

3.4 Refinements of a spherical mesh composed of eight tetrahedra and eight prisms. 142
3.5 Refinements of a portion of a circular membrane ribbon. 143
3.6 I Problem: elastic sphere subjected to normal load. 144
3.7 II Problem: hollow elastic sphere filled with acoustic fluid. 145
3.8 III Problem: acoustic fluid surrounded by PML. 146
3.9 Error indicator convergence study. 147
3.10 Equal loudness contours. Image taken from the web. 148
3.11 Base (adjacent to oval and round window), middle, and apex (at the far end of the cochlea) of the basilar membrane. The basilar membrane has variable Young modulus E. 149
3.12 Pressure distributions over basilar membrane at normal incidence. Material constants $E = 6500\,\mathrm{MPa}$, $\nu = 0.22$, $\varrho = 1412\,\mathrm{kg/m^3}$ were used for the skull. 150
3.13 Pressure distributions over basilar membrane at normal incidence. Material constants $E = 1600\,\mathrm{MPa}$, $\nu = 0.22$, $\varrho = 1412\,\mathrm{kg/m^3}$ were used for the skull. 151
3.14 Ossicles normal displacement. 152
3.15 Change in pressure ($p_{\mathrm{out}} - p_{\mathrm{in}}$) across tympanic membrane, oval window and round window. 153
3.16 Absolute value of reflection coefficient R and power transmission coefficient τ on the outer interface of the tympanic membrane. 154
3.17 Pressure on the outer (above) and inner (below) interface of the tympanic membrane. 155
3.18 Pressure distribution on base, middle and apex of basilar membrane, cf. Figure 3.11. 156
3.19 Pressure distribution oval window and round window. 157
3.20 Pressure distributions over basilar membrane at normal incidence. Ear canal was blocked. 158
3.21 Ossicles normal displacement. Ear canal was blocked. 159
3.22 Change in pressure ($p_{\mathrm{out}} - p_{\mathrm{in}}$) across tympanic membrane, oval window and round window. Ear canal was blocked. 160
3.23 Pressure on the outer (above) and inner (below) interface of the ear plug. 161
3.24 Pressure distribution on base, middle and apex of basilar membrane (cf. Figure 3.11). Ear canal was blocked. 162

A.1 Duffy's coordinates. 174
A.2 Moving systems of coordinates along edges. 187
A.3 Radial derivative reconstruction. 188
A.4 Control points of Bezier triangles. 188
A.5 Head phantom reconstruction. 189

Chapter 1

A Coupled Acoustics Elasticity Problem

1.1 Acoustics

The mathematical theory of sound propagation began with Isaac Newton (1642–1727), whose *Principia* (1686) included a mechanical interpretation of the sound as being "pressure" pulses transmitted through neighboring particles of an acoustic medium. The mathematical analysis was limited to waves of constant frequency, employed a number of crude approximations, and suffered from an incomplete definition of terminology and concepts.

Substantial progress toward the development of a viable theory of sound propagation resting on firmer mathematical and physical concepts was made during the eighteen century by Euler (1707–1783), Lagrange (1736–1813) and d'Alembert (1717–1783). During this era, continuum physics, also called field theory, evolved into a science with a definite mathematical structure. The wave equation emerged in a number of contexts, including the propagation of sound in air. The theory ultimately proposed for sound in the eighteen century was incomplete from many standpoints, however the modern theories of today can be regarded for the most part as refinements of the one developed by Euler and his contemporaries.

We start our discussion from classical fluid dynamics, following the approach presented by L.D. Landau and E.M. Lifshitz in [50]. The mathematical description of the state of a moving fluid involves the distribution of the fluid velocity $\boldsymbol{v}$, and any two thermodynamic quantities pertaining to the fluid, for example the pressure p and the density ϱ. This is a consequence of the fact that all thermodynamic quantities are determined by any two of them through the equation of state. Hence, the state of the moving fluid is completely determined by knowledge of five quantities: the three components of the velocity $\boldsymbol{v} = (v_1, v_2, v_3)$, the pressure p, and the density ϱ. Accordingly, a complete system of equations describing the dynamics of a fluid should be five in number. The first one of those equations is the equation of continuity. If the thermal conductivity and viscosity of the fluid are negligible, then the fluid is said to be an *ideal* fluid. Under this assumption, the remaining equations are the momentum equation, which is vector-valued, and the adiabatic equation. Indeed, if the assumption of absence of heat exchange between different parts of the fluid is made, then the motion of the fluid must necessarily be supposed adiabatic. The equations are:

$$\frac{d\varrho}{dt} + \varrho \operatorname{div} \boldsymbol{v} = 0 \,, \qquad \text{(Continuity Eq.)}$$

$$\varrho \frac{d\boldsymbol{v}}{dt} + \nabla p = \boldsymbol{f} \,, \qquad \text{(Momentum Eq.)}$$

$$\frac{ds}{dt} = 0 \,. \qquad \text{(Adiabatic Eq.)}$$

Here $\boldsymbol{f}$ is an external body force per unit volume, and s is the entropy per unit mass. In the context of fluid dynamics, the total derivative $d/dt = \partial/\partial t + \boldsymbol{v} \cdot \nabla$ is usually referred to as the material derivative. The conservation laws are implicitly stated in the Eulerian form.

A small amplitude oscillatory motion in a compressible fluid is called a *sound wave*. The fact that the oscillations are small implies that the velocity $\boldsymbol{v}$ is small as well. Applying the same reasoning, the change in density and pressure with respect to their value at the hydrostatic equilibrium are also small. Indeed, let's write the pressure and the density as $p = p_0 + p'$ and $\varrho = \varrho_0 + \varrho'$, where p_0 and ϱ_0 are the values at the hydrostatic equilibrium, i.e., time-independent, and p' and ϱ' are perturbations around such equilibrium, i.e., $p' \ll p_0$ and $\varrho' \ll \varrho_0$. We assume that ϱ', p', and $\boldsymbol{v}$, along with their derivatives, are of order ϵ, and neglect second order terms, in ϵ, in the continuity and momentum equations:

$$\frac{\partial \varrho'}{\partial t} + \operatorname{div}(\varrho_0\, \boldsymbol{v}) = 0 \qquad ; \qquad \varrho_0 \frac{\partial \boldsymbol{v}}{\partial t} + \nabla(p_0 + p') = \boldsymbol{f}\,. \tag{1.1.1}$$

The next step is to eliminate the density through the equation of state

$$p = P(\varrho, s)\,. \tag{State Eq.}$$

Such an equation, contrary to the conservation equations, is implicitly defined on the material points of the the fluid, i.e., in a Lagrangian frame. By virtue of the Adiabatic Equation, we have that

$$\frac{dp}{dt} = \frac{\partial P}{\partial \varrho}\frac{d\varrho}{dt} + \frac{\partial P}{\partial s}\frac{ds}{dt} = \frac{\partial P}{\partial \varrho}\frac{d\varrho}{dt}\,.$$

By setting $c_0^2 := \dfrac{\partial P}{\partial \varrho}(\varrho_0, s_0)$, which is *a posteriori* interpreted as the intrinsic sound speed in the acoustic medium, and discarding higher order terms in ϵ, we obtain

$$\frac{\partial p'}{\partial t} + \boldsymbol{v} \cdot \nabla p_0 = c_0^2 \left(\frac{\partial \varrho'}{\partial t} + \boldsymbol{v} \cdot \nabla \varrho_0 \right).$$

Solving for $\partial\varrho'/\partial t$ and substituting in the first one of equations (1.1.1) yields:

$$\frac{1}{c_0^2}\left(\frac{\partial p'}{\partial t}+\boldsymbol{v}\cdot\nabla p_0\right)+\varrho_0\,\mathrm{div}\,\boldsymbol{v}=0\,.$$

Under the assumption that body forces, e.g., gravity may be neglected, the ambient pressure p_0 is constant, and the equations (1.1.1) become:

$$\frac{1}{c_0^2}\frac{\partial p}{\partial t}+\varrho_0\,\mathrm{div}\,\boldsymbol{v}=0 \qquad ; \qquad \varrho_0\frac{\partial \boldsymbol{v}}{\partial t}+\nabla p=\boldsymbol{f}\,, \tag{1.1.2}$$

where we have dropped the primes for succinctness. The linearized continuity and momentum equations are applicable under the condition that $|\boldsymbol{v}|\ll c_0$. This condition can be obtained as a consequence of the assumption $\varrho'\ll\varrho_0$. Indeed, when considering traveling plane waves, it is easy to derive the relation $v=p'/(\varrho c_0)$, which implies that $v=(\varrho'/\varrho)c_0$.

We shall restrict ourself to the class of time-harmonic problems, that is

$$p'(x,y,z;t)=\Re\big[\hat{p}(x,y,z)\,e^{i\omega t}\big] \qquad ; \qquad \boldsymbol{v}(x,y,z;t)=\Re\big[\hat{\boldsymbol{v}}(x,y,z)\,e^{i\omega t}\big]\,, \tag{1.1.3}$$

where $\hat{p}$ and $\hat{\boldsymbol{v}}$ are *a priori* complex quantities and ω is a given frequency. Let us consider the first one of equations (1.1.3). Using the fact that the derivative with respect to time or space commutes with the operation of taking the real part, and that the product of a real number with the real part of a complex number is the real part of the product, we obtain that:

$$\begin{aligned}0&=\frac{1}{c_0^2}\frac{\partial\Re(\hat{p}e^{i\omega t})}{\partial t}+\varrho_0\,\mathrm{div}\,\Re(\hat{\boldsymbol{v}}e^{i\omega t})=\\&=\frac{1}{c_0^2}\Re\frac{\partial(\hat{p}e^{i\omega t})}{\partial t}+\varrho_0\Re\,\mathrm{div}(\hat{\boldsymbol{v}}e^{i\omega t})=\Re\left[\left(\frac{i\omega}{c_0^2}\hat{p}+\varrho_0\,\mathrm{div}\,\hat{\boldsymbol{v}}\right)e^{i\omega t}\right]\end{aligned}$$

A sufficient condition for the previous chain of equality to hold is that the quantity in parenthesis in the right-hand-side vanishes. A similar reasoning holds for the second one of equations (1.1.3). Dropping the karats we obtain the following first order system:

$$\frac{i\omega}{c_0^2} p + \varrho_0 \operatorname{div} \boldsymbol{v} = 0 \,, \tag{1.1.4}$$

$$i\omega \varrho_0 \boldsymbol{v} + \nabla p = \boldsymbol{f} \,. \tag{1.1.5}$$

An alternative approach relies on the use of the Fourier transform. Let $\hat{u}(\omega) = \big[\mathscr{F}_{t\to\omega}\, u\big](\omega)$ be the Fourier transform with respect to time. By proceeding *formally* on equations (1.1.2), and using the fundamental property $\big[\mathscr{F}_{t\to\omega}\, \partial u/\partial t\big](\omega) = i\omega\, \hat{u}(\omega)$, we obtain the above system of PDE's, with p, $\boldsymbol{v}$, and $\boldsymbol{f}$ replaced by their respective Fourier transforms. The acoustic pressure and elastic displacement are recovered by applying the inverse Fourier transform. Let us recall that, in the case of a real function u, the Fourier transform is Hermitian, i.e., $\hat{u}(-\omega) = \overline{\hat{u}(\omega)}$. This implies that:

$$\begin{aligned} u = \mathscr{F}^*_{\omega\to t}\, \hat{u} = \int_{-\infty}^{+\infty} \hat{u}(\omega)\, e^{i\omega t}\, d\omega = \int_{-\infty}^{0} \hat{u}(\omega)\, e^{i\omega t}\, d\omega + \int_{0}^{+\infty} \hat{u}(\omega)\, e^{i\omega t}\, d\omega = \\ = \int_{0}^{+\infty} \overline{\hat{u}(\omega)\, e^{i\omega t}}\, d\omega + \int_{0}^{+\infty} \hat{u}(\omega)\, e^{i\omega t}\, d\omega = \\ = 2\int_{0}^{+\infty} \Re\big[\hat{u}(\omega)\, e^{i\omega t}\big]\, d\omega = \int_{-\infty}^{+\infty} \Re\big[\hat{u}(\omega)\, e^{i\omega t}\big]\, d\omega \,, \end{aligned}$$

where we have used the fact that the function $\Re\big[\hat{u}(\omega)\, e^{i\omega t}\big]$ is symmetric with respect to the origin. In the case of a signal with just one angular frequency component $\omega = \omega_0$, we obtain that $u = \Re\big[\hat{u}(\omega_0)\, e^{i\omega_0 t}\big]$. The discussion can be made rigorous by introducing a sequence of distributions $\{\hat{u}_\varepsilon\}_{\varepsilon>0}$ converging to a delta distribution δ_{ω_0}.

In the case of unforced waves, i.e., $\boldsymbol{f} = 0$, the above system of first order equations can be reduced to one second order PDE by taking the divergence of equation (1.1.5) and substituting into equation (1.1.4):

$$-k^2 p - \Delta\, p = 0\ .$$

This is the well-known Helmholtz, equation; $k = \omega/c_0$ is the wave number.

When considering time-harmonic wave propagation in free space, namely in an unbounded acoustic domain Ω^c, where Ω is the acoustic scatterer, we *postulate* that no waves are reflected from infinity. The mathematical expression of this far field condition is the well known Sommerfeld condition and can be immediately obtained through the free space Green's function. It is well known, see e.g. L.C. Evans [34], that in the three dimensional case, $G(x,y) = e^{-ik|x-y|}/4\pi|x-y|$, where x and y are two points in 3D space, is the fundamental radiating—recall that we picked time ansatz $e^{i\omega t}$!—solution of the Helmholtz equation. Then, a solution p is written in terms of the Green's function as

$$p(x) = \int_{\partial\Omega} \left[p(y)\frac{\partial G(x,y)}{\partial \boldsymbol{n}} - G(x,y)\frac{\partial p(y)}{\partial \boldsymbol{n}} \right] dS(y)\ .$$

We truncate the outer domain Ω^c at some far-field boundary using a sphere B_R with a sufficiently large radius. If no waves are to be reflected from infinity, then the contribution to the boundary integral coming from ∂B_R must vanish as $R \to \infty$. Whenever R is big enough, we can set $r = |x-y|$ and replace the partial derivative in the normal direction by the derivative in the radial direction, to obtain:

$$\left| \int_{\partial B_R} \frac{1}{r}\Big(-ikp - \frac{p}{r} - \frac{\partial p}{\partial r}\Big) \frac{e^{ikr}}{4\pi} dS \right| \leq \int_{\partial B_R} \left| \frac{p}{4\pi r^2} \right| dS + \int_{\partial B_R} \frac{1}{4\pi r} \left| ikp + \frac{\partial p}{\partial r} \right| dS \, .$$

Thus, the integrals vanish if the following conditions hold:

$$p = \mathrm{O}(1/r) \qquad ; \qquad ikp + \frac{\partial p}{\partial r} = \mathrm{o}(1/r) \quad (\text{as } r \uparrow \infty) \, . \tag{1.1.6}$$

The first condition describes the decay of the solution, and the second one its directional character. Since we are interested in finite energy solutions, the decay condition is already satisfied and we are left with the Sommerfeld radiation condition, namely the second one of the conditions (1.1.6). In practice, we eliminate incoming waves through the use of a Perfectly Matched Layer, see Section 1.5.

Let us discuss the variational formulation for the Helmholtz equation in an unbounded domain. Let us assume a Neumann boundary condition on $\partial\Omega$, i.e., $\nabla p \cdot \boldsymbol{n} = g$, and enclose the scatterer with a sufficiently large truncating sphere B_R. We rewrite the Sommerfeld condition as

$$\frac{\partial p}{\partial r} = -ikp + \varphi \, ,$$

where $\varphi = \mathrm{O}(1/r^2)$ is an unknown function. Multiplying the Helmholtz equation by the complex conjugate of a test function q and integrating by parts over the annular domain $D_R := B_R \setminus \Omega$, we obtain

$$\int_{D_R} \big(\nabla p \cdot \nabla \overline{q} - k^2 p \, \overline{q} \big) - ik \int_{\partial B_R} p \, \overline{q} = \int_{\partial\Omega} g \, \overline{q} + \int_{\partial B_R} \varphi \, \overline{q} \, .$$

We need to study the behavior as $R \to \infty$. By separation of variables, it is possible to show that the leading term of p is of the form e^{-ikr}/r, i.e., a spherical Hankel function of the second kind, hence both p and ∇p are not integrable over the external domain Ω^c. A remedy is to employ test functions that are of order $\mathrm{O}(1/r^3)$. However, if this were the case, then both boundary integrals over ∂B_R would vanishing, making it impossible to build the Sommerfeld condition right into the variational form. A solution, proposed by Leis, see [54], is to resort to weighted Sobolev spaces. Indeed, following the discussion in Gerdes and Demkowicz, see [41], let us define:

$$H^{1,w}(\Omega^c) = \{q \,:\, \|q\|_{1,w} < \infty\}\,,$$

with the norm $\|\cdot\|_{1,w}$ corresponding to the inner product:

$$(p,q)_{1,w} = \int_{\Omega^c} \big(wp\,\overline{q} + w\nabla p\cdot\nabla\overline{q}\big) + \int_{\Omega^c} \left(\frac{\partial p}{\partial r} + ikp\right)\overline{\left(\frac{\partial q}{\partial r} + ikq\right)}\,.$$

Two particular weights are of interest: $w = 1/r^2$ and $w^* = r^2$. Then the variational formulation becomes:

$$\begin{aligned}&\text{Find } p \in H^{1,w}(\Omega^c) \text{ such that:}\\ &\int_{\Omega^c}\big(\nabla p\cdot\nabla\overline{q} - k^2 p\,\overline{q}\big) = \int_{\partial\Omega} g\,\overline{q} \qquad \forall\, q \in H^{1,w^*}(\Omega^c)\,.\end{aligned}$$

In practice, see Section 1.5, we will reduce ourselves to a bounded computational through a technique called Perfectly Matched Layer.

In the previous discussion we have completely neglected the effect of energy dissipation occurring during the motion of a viscous fluid. If thermal effects were to be included, they would lead to a different equation of motion and a different equation for the entropy. This is the starting point for the development of non linear acoustics, a field that is experiencing a fast growth, specially because of its biomedical applications. Let us state the equation of motion, the well known Navier-Stokes equation, and entropy equation without discussing their derivation:

$$\varrho \frac{d\boldsymbol{v}}{dt} + \nabla p = \eta \Delta \boldsymbol{v} + \big(\zeta + \tfrac{1}{3}\eta\big) \nabla \operatorname{div} \boldsymbol{v} \,,$$
$$\varrho T \frac{ds}{dt} = \operatorname{div}(\kappa \, \nabla T) + \tfrac{1}{2}\eta \left(\frac{\partial v_i}{\partial x_k} + \frac{\partial v_k}{\partial x_i} - \tfrac{2}{3}\delta_{ik} \frac{\partial v_l}{\partial x_l} \right)^2 + \zeta (\operatorname{div} \boldsymbol{v})^2 \,.$$

In the entropy equation we have used index notation, namely summation over repeated indices is understood. Here η is the shear viscosity, ζ is the bulk viscosity, κ is the thermal conductivity, and T is the temperature. After a number of non trivial manipulations, see M. Hamilton and D. Blackstock [43], we obtain the following system:

$$\frac{\partial \varrho}{\partial t} + \varrho_0 \operatorname{div} \boldsymbol{v} = \frac{1}{\varrho_0 c_0^4} \frac{\partial p^2}{\partial t} + \frac{1}{c_0^2} \frac{\partial \mathcal{L}}{\partial t} \,,$$
$$\varrho_0 \frac{\partial \boldsymbol{v}}{\partial t} + \nabla p = -\frac{1}{\varrho_0 c_0^2} \big(\zeta + \tfrac{4}{3}\eta\big) \nabla \frac{\partial p}{\partial t} - \nabla \mathcal{L} \,,$$

where $\mathcal{L} = \frac{1}{2}\varrho_0 |\boldsymbol{v}|^2 - p^2/2\varrho_0 c_0^2$ is the so-called second order Lagrangian density.

1.2 Elasticity

The discussion on elasticity follows closely the one by L.D. Landau and E.M. Lifshitz in [51]. In this section we consistently use index notation. We introduce the concept of stress and strain, show that the stress tensor must be symmetric, state Hooke's law, and derive the equations of motion for an elastic body.

When an elastic deformation of a body occurs, forces which tend to return the body to its equilibrium position arise. These forces are called internal stresses. Let us consider the total force acting on some portion of the body, which can be written as $\int \boldsymbol{F}\, dV$, where $\boldsymbol{F}$ is a force per unit volume. The forces with which various parts of the portion considered act on one another give a zero resultant force by Newton's third law. Hence, the sought total force can be regarded as the sum of the forces exerted on that portion of the body by the portions surrounding it. These forces act on the surface of the portion, therefore the volume integral can be reduced to a surface integral. If this is the case, there exists a rank two tensor σ_{ik}, called the *stress* tensor, such that $F_i = \partial\sigma_{ik}/\partial x_k$. Notice that $\boldsymbol{F}$ includes inertial forces as well. Since the portion is arbitrary, such relation has to hold for each point in the body.

The moment of the force $\boldsymbol{F}$ with respect to the origin can be written as an antisymmetric tensor $F_i x_k - F_k x_i$. This is a consequence of the fact that the cross product is a pseudo-vector. The total moment acting on a portion of a body is $M_{ik} = \int (F_i x_k - F_k x_i)\, dV$ and a reasoning similar to the previous one implies that such a quantity can be reduced to an integral over the surface. Indeed, using Gauss theorem we have that:

$$M_{ik} = \int \left(\frac{\partial\sigma_{il}}{\partial x_l} x_k - \frac{\partial\sigma_{kl}}{\partial x_l} x_i \right) dV =$$

$$\int \frac{\partial(\sigma_{il}x_k - \sigma_{kl}x_i)}{\partial x_l}\, dV - \int (\sigma_{ik} - \sigma_{ki})\, dV = \oint (\sigma_{il}x_k - \sigma_{kl}x_i)\, df_l - \int (\sigma_{ik} - \sigma_{ki})\, dV\,,$$

where df_l are the components of the surface element vector. If M_{ik} has to be an integral over the surface, then the second term of last equality must vanish. This implies that the stress tensor is symmetric, i.e., $\sigma_{ik} = \sigma_{ki}$.

The displacement due to a deformation is given by $u_i = x'_i - x_i$, where x_i are the coordinate of a point in the undeformed body and x'_i are the coordinates of the displaced point. Indeed the displacement is a function of x_i. Let us consider two points in the undeformed body apart from each other by an infinitesimal vector dx_i. The distance between such points before the deformation occurs is $dl = [(dx_1)^2 + (dx_2)^2 + (dx_3)^2]^{1/2}$, while afterwards it is $dl' = [(dx'_1)^2 + (dx'_2)^2 + (dx'_3)^2]^{1/2}$. Using the fact that $dx'_i = dx_i + du_i$ and $du_i = (\partial u_i/\partial x_k)dx_k$, we obtain:

$$dl'^2 = dl^2 + 2\frac{\partial u_i}{\partial x_k}\,dx_i dx_k + \frac{\partial u_i}{\partial x_k}\frac{\partial u_i}{\partial x_l}\,dx_k dx_l = dl^2 + 2\varepsilon_{ik}\,dx_i dx_k\ .$$

In the above expression we have used some index manipulation and defined the *strain* tensor ε_{ik} as:

$$\varepsilon_{ik} = \tfrac{1}{2}\left(\frac{\partial u_i}{\partial x_k} + \frac{\partial u_k}{\partial x_i} + \frac{\partial u_l}{\partial x_i}\frac{\partial u_l}{\partial x_k}\right)\ .$$

The strain tensor is a measure of local change in volume and shape. In the case of small deformation, the second order term is neglected and we obtain $\varepsilon_{ik} = \frac{1}{2}(\partial u_i/\partial x_k + \partial u_k/\partial x_i)$, i.e., the strain is equal to the symmetric part of the displacement gradient. By diagonalising the strain tensor and recalling that the trace is invariant under such a transformation, it is seen that the change in volume in the deformation is given by ε_{ii}. Whenever this sum is zero, the volume of the body is unchanged and the deformation is called a *pure shear*. On the other hand, a transformation of the type $\alpha\delta_{ij}$, where α is and arbitrary constant, will cause a change in volume while preserving the shape. This is called a *hydrostatic compression*.

In the case of an isotropic material, the stress tensor is related to strain tensor by Hooke's law. Hooke's law is derived using thermodynamic arguments and states that:

$$\sigma_{ik} = K\varepsilon_{ll}\delta_{ik} + 2\mu\big(\varepsilon_{ik} - \tfrac{1}{3}\delta_{ik}\varepsilon_{ll}\big)\ .$$

The quantities K and $\mu > 0$ are called, respectively, the *bulk modulus* and the *shear modulus*. The previous expression shows that, if the deformation is a pure shear or hydrostatic compression, the relationship between σ_{ik} and ε_{ik} is determined only be the shear modulus or by the bulk modulus respectively. The bulk modulus is related to the Lamé coefficients by $K = \lambda + \frac{2}{3}\mu$. While the shear modulus μ is a positive quantity, the first Lamé parameter λ can in principle be negative. However, for typical elastic materials, λ should be non negative. As we will see, the sign of λ and μ is crucial for proving the coercivity of the elasticity operator.

The equation of motion for an elastic medium is readily obtained as

$$\frac{\partial^2}{\partial t^2}(\varrho u_i) = \frac{\partial \sigma_{ik}}{\partial x_k} + f_i \ ,$$

where ϱ is the density and f_i are the external forces per unit volume. We again restrict ourself to the case of time-harmonic problems, i.e., $u_i = \Re\,\hat{u}_i e^{i\omega t}$, $\sigma_{ij} = \Re\,\hat{\sigma}_{ij} e^{i\omega t}$, $f_i = \Re\,\hat{f}_i e^{i\omega t}$. Assuming ϱ to be constant, and applying the same reasoning of the discussion on acoustics, we obtain:

$$-i\omega^2 \varrho u_i = \frac{\partial \sigma_{ik}}{\partial x_k} + f_i \ ,$$

where we have dropped the hats for succinctness. Through the definition of strain we relate the stress tensor to the gradient of the deformation and obtain the following system:

$$\sigma_{ik} = \lambda \frac{\partial u_l}{\partial x_l}\delta_{ik} + \mu\left(\frac{\partial u_i}{\partial x_k} + \frac{\partial u_k}{\partial x_i}\right) = E_{ijkl}\frac{\partial u_j}{\partial x_l} \ ,$$

where $E_{ijkl} = \mu(\delta_{ik}\delta_{jl} + \delta_{il}\delta_{jk}) + \lambda\delta_{ij}\delta_{kl}$ is called the tensor of elasticities and enjoys the minor symmetries $E_{ijkl} = E_{jikl}$, $E_{ijkl} = E_{ijlk}$ and major symmetry $E_{ijkl} = E_{klij}$.

1.3 Coupling Acoustics and Elasticity

Rigid scattering of sound happens when an acoustical wave hits a rigid obstacle. As we will see, in this case the wave is totally reflected and no energy is transmitted to the obstacle. If the obstacle is elastic, the wave is partially reflected and partially transmitted in the form of elastic vibrations. More specifically, the acoustical pressure waves act as a time-varying load on the elastic body. In this case we speak of elastic scattering. On the other hand, whenever an acoustical medium picks up vibrations of an embedded body in the form of acoustical waves, we say that the sound is radiated from the body. A typical example are pressure waves generated by a transducer, namely a pulsating sphere. This section is dedicated to the development of the mathematical formulation for the coupled problem.

Let Ω be a bounded domain in $\mathbb{R}^3$, which is split into two disjoint parts: a subdomain Ω_e occupied by an elastic medium, and a subdomain Ω_a occupied by an acoustical fluid. In precise mathematical terms, we require Ω, Ω_a and Ω_e to be open sets such that $\Omega_a \cap \Omega_e = \emptyset$ and $\overline{\Omega} = \overline{\Omega}_a \cup \overline{\Omega}_e$. In the elastic subdomain the quantities of interest are the stress tensor $\boldsymbol{\sigma}$ and the elastic displacement $\boldsymbol{u}$, while in the acoustic subdomain the quantities of interest are the acoustic pressure p and the particle velocity $\boldsymbol{v}$. The two subdomains are separated by an interface $\Gamma_I = (\overline{\Omega}_a \cap \overline{\Omega}_e) \cap \Omega$. Neither the subdomains nor the interface need to be connected. The coupling will be obtained by imposing appropriate interface conditions on the interface Γ_I. For an ideal fluid, such as the acoustical fluid, the boundary condition for the equation of motion involves the normal component of the velocity, v_n. We require that the normal component of the velocity of the acoustical fluids matches the normal component of the velocity of the elastic medium. Taking advantage of the fact that the elastic deformation is time-harmonic we have:

$$\boldsymbol{v} \cdot \boldsymbol{n} = i\omega\, \boldsymbol{u} \cdot \boldsymbol{n}\,, \tag{1.3.7}$$

where $\boldsymbol{n}$ is either the outward normal of the acoustic or the elastic subdomain, as long as it is consistent on both sides of the equality. On the other hand, the boundary condition for the equation of elastic deformation prescribes the traction $\boldsymbol{\sigma n}$. In order to use such a condition, we need to develop the notion of stress tensor for a fluid. Let us suppose that the only force acting on the fluid is the one due to pressure, and that the fluid is in hydrostatic equilibrium. The force exerted on a region of the fluid by the fluid surrounding it is $-\oint p\, df_i$. By definition of stress tensor σ_{ik}, the previous quantity must be equal to $\oint \sigma_{ik}\, df_k$, which implies that $\sigma_{ik} = -p\delta_{ik}$. Hence the second interface condition is:

$$-p\boldsymbol{n} = \boldsymbol{\sigma n}\,. \tag{1.3.8}$$

The two interface conditions constitute a *weak* coupling. This expression refers to a number of things. The involved variables, elastic displacement $\boldsymbol{u}$ and acoustic pressure p, are not forced to match each other on the elastic/acoustic interface. Instead, the variational formulation incorporates additional off-diagonal flux terms representing the interaction in between the two fields and resulting from the interface conditions. It is essential that either field represents a dual quantity for the other. Hence the term *weak* contrary to *strong* couplings where two variables match each other (in terms of traces) at both continuous and discrete levels along the interface. A second meaning of the term weak coupling refers to the fact that the coupling terms represent compact perturbations of the terms on the main diagonal corresponding to elasticity and acoustics. This guarantees automatically that the Galerkin discretization of the problem will be asymptotically stable. We anticipate that the variational formulation for the coupled problem that we are about to derive is classical and has been extensively studied over the last two decades, see e.g. [39].

In order to introduce boundary conditions, the external boundary $\partial\Omega$ will be partitioned into Dirichlet, Neumann and Cauchy parts: $\Gamma_D, \Gamma_N, \Gamma_C$, respectively. Each of these boundary parts may consist of a part belonging to the boundary $\partial\Omega_e$ of the elastic subdomain, or the boundary $\partial\Omega_a$ of the acoustical subdomain. They need not to be connected. We introduce the following boundary conditions:

(i) Dirichlet boundary condition:

$$p = p_D \qquad \text{on } \Gamma_D \cap \partial\Omega_a \qquad ; \qquad u_i = u_{i,D} \qquad \text{on } \Gamma_D \cap \partial\Omega_e \; .$$

(ii) Neumann boundary condition:

$$v_n = v_0 \qquad \text{on } \Gamma_N \cap \partial\Omega_a \qquad ; \qquad \sigma_{ik} n_k = g_i \qquad \text{on } \Gamma_N \cap \partial\Omega_e \; .$$

(iii) Cauchy boundary condition with impedance constant $d > 0$:

$$v_n = dp + v_0 \qquad \text{on } \Gamma_C \cap \partial\Omega_a \qquad ; \qquad \sigma_{ik} n_k + i\omega\beta_{ik} u_k = g_i \qquad \text{on } \Gamma_C \cap \partial\Omega_e \; .$$

Let's discuss the physical meaning of the boundary conditions, starting with acoustics. A forward traveling wave $p = f(t - \boldsymbol{n} \cdot \boldsymbol{x}/c_0)$ of arbitrary waveform $f(\cdot)$

generates, according to the momentum equation, a particle velocity $\boldsymbol{v} = \boldsymbol{n}/(\varrho_0 c_0)\, p$, hence $p/v_n = \varrho_0 c_0$. The quantity $Z_0 = \varrho_0 c_0$ is called *characteristic impedance* of the fluid. In general, when p is not necessarily a traveling wave, the quantity p/v_n is a complex quantity called *specific acoustic impedance* and indicated by Z_s. The real part $\Re\, Z_s$ is called *specific acoustic resistance*, while the imaginary part $\Im\, Z_s$ is called *specific acoustic reactance*. Let p^{inc} be an incident pressure wave on the interface between two semi-infinite media. At the interface the wave is partly reflected and partly transmitted: $p^{\text{inc}} + p^{\text{ref}} = p^{\text{tr}}$. The case $Z_s = 0$ implies a Dirichlet condition for the pressure, $p = 0$. This corresponds to a soft boundary or pressure release interface where $p^{\text{ref}} = -p^{\text{inc}}$ and $p^{\text{tr}} = 0$. Hence no pressure wave is transmitted and the reflected wave has opposite polarization then the incident one. On the other hand, the case $|Z_s| = \infty$ corresponds to a hard wall. It implies a Neumann boundary condition for the velocity expressed in terms of pressure, $v_n = 0$. At the interface the following condition for the particle velocity holds $\boldsymbol{v}^{\text{inc}} + \boldsymbol{v}^{\text{ref}} = \boldsymbol{v}^{\text{tr}}$, which translates into $v_n^{\text{tr}} = 0$ and $v_n^{\text{inc}} + v_n^{\text{ref}} = 0$. Again, no signal is transmitted beyond the interface. Using the characteristic impedance to switch from velocity to pressure and recalling that the relationship for a backward traveling wave carries a minus sign, we find that $p^{\text{ref}} = p^{\text{inc}}$ at the interface. The reflected pressure wave is an exact replica of the incident wave and pressure doubling occurs at the wall.

The physical meaning of boundary conditions for elasticity is immediate. The Dirichlet condition is a prescribed displacement, while the Neumann condition is a prescribed traction. For example, the case of the fixed-free cantilever beam corresponds to a homogeneous Dirichlet condition at the fixed end and a homogeneous

Neumann condition at the free end. In the case of an impedance boundary condition, the traction is set equal to a load and a force proportional to the velocity, as in the case of a dashpot.

We proceed with the derivation of the variational formulation for the coupled problem. Let's discuss acoustics first. We start with the weak form of the continuity equation for acoustics and integrate by parts. Using the momentum equation, and the boundary and interface conditions, we obtain

$$
\begin{aligned}
& p = p_D \text{ on } \Gamma_{D,a} \\
& \int_{\Omega_a} \left(\frac{i\omega}{c^2} pq + \frac{1}{i\omega} \nabla p \cdot \nabla q \right) + \int_{\Gamma_{C,a}} \varrho_f d\, pq + \int_{\Gamma_I} i\omega \varrho_f u_n\, q = \int_{\Gamma_{N,a} \cup \Gamma_{C,a}} \varrho_f v_0 q \\
& \hspace{20em} \forall q \ : \ q = 0 \text{ on } \Gamma_{D,a} \, .
\end{aligned}
$$

In the case of elasticity, we start from the weak form of the conservation of momentum and integrate by parts. Using the constitutive relation, and the boundary and interface conditions, we find

$$
\begin{aligned}
& u = u_{i,D} \text{ on } \Gamma_{D,e} \\
& \int_{\Omega_e} \left(E_{ijkl} u_{k,l} v_{i,j} - \varrho_s \omega^2 u_i v_i \right) + i\omega \int_{\Gamma_{C,e}} \beta_{ij} u_j v_i + \int_{\Gamma_I} p v_n = \int_{\Omega_e} f_i v_i + \int_{\Gamma_{N,e} \cup \Gamma_{C,e}} g_i v_i \\
& \hspace{20em} \forall v_i \ : \ v_i = 0 \text{ on } \Gamma_{D,e} \, .
\end{aligned}
$$

where $E_{ijkl} = \mu(\delta_{ik}\delta_{jl} + \delta_{il}\delta_{jk}) + \lambda \delta_{ij}\delta_{kl}$. We multiply the variational statement for acoustics by factor $i\omega$ and define the acoustic wave number $k = \omega/c$ in order to get the final variational formulation for the coupled problem. We define spaces of test functions

$$
U = \left\{ \boldsymbol{v} \in H^1(\Omega_e)^3 \ : \ \boldsymbol{v} = \boldsymbol{0} \text{ on } \Gamma_{D,e} \right\} \qquad ; \qquad V = \left\{ q \in H^1(\Omega_a) \ : \ q = 0 \text{ on } \Gamma_{D,a} \right\} ,
$$

and let $\tilde{\boldsymbol{u}}_D \in H^1(\Omega_e)^3$ be a finite energy lifting of displacements $\boldsymbol{u}_D$ prescribed on $\Gamma_{D,e}$ and $\tilde{p}_D \in H^1(\Omega_a)$ be a a finite enery lift of pressure p_D prescribed on $\Gamma_{D,a}$. The final variational formulation becomes: *Find* $\boldsymbol{u} \in \tilde{\boldsymbol{u}}_D + U$, $p \in \tilde{p}_D + V$ *such that:*

$$\begin{array}{lclcll} b_{ee}(\boldsymbol{u},\boldsymbol{v}) & + & b_{ae}(p,\boldsymbol{v}) & = & l_e(\boldsymbol{v}) & \forall \boldsymbol{v} \in U\ , \\ b_{ea}(\boldsymbol{u},q) & + & b_{aa}(p,q) & = & l_a(q) & \forall q \in V\ . \end{array} \tag{1.3.9}$$

The bilinear and linear forms are given by the formulas:

$$b_{ee}(\boldsymbol{u},\boldsymbol{v}) = \int_{\Omega_e} \left(E_{ijkl} u_{k,l} v_{i,j} - \varrho_s \omega^2 u_i v_i \right) + i\omega \int_{\Gamma_{C,e}} \beta_{ij} u_j v_i\ , \tag{1.3.10a}$$

$$b_{ae}(p,\boldsymbol{v}) = \int_{\Gamma_I} p\, v_n\ , \tag{1.3.10b}$$

$$b_{ea}(\boldsymbol{u},q) = -\,\omega^2 \varrho_f \int_{\Gamma_I} u_n\, q\ , \tag{1.3.10c}$$

$$b_{aa}(p,q) = \int_{\Omega_a} \left(\nabla p \cdot \nabla q - k^2 p\, q \right) + i\omega \int_{\Gamma_{C,a}} \varrho_f d\, p\, q\ , \tag{1.3.10d}$$

$$l_e(\boldsymbol{v}) = \int_{\Omega_e} f_i v_i + \int_{\Gamma_{N,e} \cup \Gamma_{C,e}} g_i v_i\ , \tag{1.3.10e}$$

$$l_a(q) = i\omega \varrho_f \int_{\Gamma_{N,a} \cup \Gamma_{C,a}} v_0 q\ . \tag{1.3.10f}$$

Coupled problem (1.3.9) is symmetric if and only if diagonal forms b_{ee} and b_{aa} are symmetric and $b_{ae}(p,\boldsymbol{u}) = b_{ea}(\boldsymbol{u},p)$. Thus, in order to enable the symmetry of the formulation, we need to rescale problem by, for instance, dividing the second equation by factor $\omega^2 \varrho_f$. Note that we refer to the outer normal unit vector $\boldsymbol{n}$ always *locally*, i.e., in the formula for the coupling bilinear form b_{ae} involving elasticity test functions $\boldsymbol{v}$, versor $\boldsymbol{n}$ points outside of the elastic domain, whereas in the formula for the coupling bilinear form b_{ea} involving acoustic test functions q, versor $\boldsymbol{n}$ points outside of the acoustic domain. The normal components v_n and u_n present in the coupling terms are thus opposite to each other, and the formulation is indeed symmetric.

1.4 Mathematical Analysis

The problem of interest falls into the class of compact perturbation of the identity. The main theoretical result is the Fredholm alternative, that we briefly discuss. Let X, Y be Banach spaces and let L be a linear and continuous operator from X into Y, such that $L = A + C$, where A is an isomorphism and C is compact. An isomorphism between normed vector spaces is a continuous linear bijection whose inverse is continuous as well. Thus, we can equivalently consider the operator $L = I + K$, where I is the identity in X and $K = A^{-1}C$ is still a compact operator by virtue of the compactness of C. It is well known that for an operator L of this type the range $R(L)$ is closed and the kernel $N(L)$ has finite dimension. This result extends to operators of the type $L = K - \lambda I$ for some $\lambda \neq 0$ by rewriting $K - \lambda I = \lambda(\lambda^{-1}K - I)$ and recalling that the space $\mathcal{K}(X)$ of compact operators on X, being is a vector space, is in particular closed with respect to scalar multiplication. Thus, the previous result implies that $N(K - \lambda I)$, i.e., the eigenspace associated to λ, has finite dimension and $(K - \lambda I)X$ is a closed subspace of X. The proof of the Fredholm alternative relies on the Riesz lemma.

Lemma 1.4.1 (Riesz). *If Z is a closed proper subspace of a normed linear space X, and if $0 < \lambda < 1$, then there exists a point $x \in X$ such that $\|x\| = 1$ and* $\operatorname{dist}(x, Z) \geq \lambda$.

Proof. Take $x_0 \in X \setminus Z$ and set $\delta = \operatorname{dist}(x_0, Z) > 0$; since $\delta/\lambda > \delta$, there exists $z \in Z$ such that $\|x_0 - z\| \leq \delta/\lambda$. Set $y = x_0 - z$, then $y \neq 0$ and set $\alpha = \|y\| > 0$

and $x = y/\|y\|$. For every $z' \in Z$, $y + \alpha z' \in Z$ and

$$\|x - z'\| = \frac{1}{\alpha}\|y - \alpha z'\| = \frac{1}{\alpha}\|x_0 - z - \alpha z'\| \geq \frac{\delta}{\alpha} = \frac{\delta}{\|x_0 - z'\|} \geq \frac{\delta}{\delta/\lambda} = \lambda \,.$$

Since z' is arbitrary, then $\operatorname{dist}(x, Z) \geq \lambda$. ■

In simple terms, the Fredholm alternative states that for an operator $L = K - \lambda I$ the following alternative holds:

either L is a an isomorphism, *or* L is not injective.

This result is an immediate consequence of the fact that the spectrum of a compact operator reduces to the point spectrum, that is the set of the eigenvalues, and, when in infinite dimension, the set $\{0\}$. Here is the precise statement.

Theorem 1.4.2 (Fredholm alternative)**.** *Let X be a Banach space and $K \in \mathcal{K}(X)$. If $\lambda \neq 0$, then λ is either an eigenvalue of K, or $K - \lambda I$ has a bounded inverse.* ■

Proof. Note that if $T : X \to X$ is any operator, then $TX \subset X$ therefore we recursively obtain that $T^{n+1}X \subset T^n X$. Let's set $X_0 = X$ and define recursively $X_{n+1} = (K - \lambda I)X_n$. We obtain the nested sequence of closed subspaces $X_0 \supset X_1 \supset X_2 \supset \cdots$. As long as $X_n \neq X_{n+1}$, by the Riesz lemma we can construct $x_n \in X_n$ such that $\|x_n\| = 1$ and $\operatorname{dist}(x_n, X_{n+1}) \geq 1/2$. Suppose $n < m$ and pick $\lambda \neq 0$, then

$$Kx_n - Kx_m = \lambda x_n + \left[(K - \lambda I)x_n - \lambda x_m - (K - \lambda I)x_m\right] ,$$

where the term in square brackets indeed belongs to X_{n+1}. Let's call such term $-\lambda x$, for some $x \in X_{n+1}$. We have that

$$\|Kx_n - Kx_m\| = |\lambda|\|x_n - x\| \geq |\lambda|/2 > 0 \,.$$

Since K is compact, the sequence $\{Kx_n\}$ admits a convergent subsequence. Thus, the previous construction ought to stop somewhere, i.e., there exists n such that $X_n = X_{n+1}$. Thus, $K - \lambda I : X_n \to X_{n+1}$ is a surjection. Assume $K - \lambda I$ to be injective as well. Pick $y \in X$ and construct $y^{n+1} = (K-\lambda I)^{n+1} y \in X_{n+1}$. Then there exists $x^n \in X_n$ such that $(K - \lambda I)x^n = y^{n+1}$, and $x^n = (K - \lambda I)x$ for some $x \in X$. Therefore, for every $y \in X$, there exists $x \in X$ such that $(K-\lambda I)^{n+1}x = (K-\lambda I)^n y$. Such a relation can be rewritten as:

$$(K - \lambda I)^n \big[(K - \lambda I)x - y\big] = 0 \, .$$

Since $(K-\lambda I)^n$ is injective as well, the term in square brackets must vanish, therefore $(K-\lambda I)x = y$. This proves that $K-\lambda I$ is also surjective. Finally, the open mapping theorem assures that $(K - \lambda I)^{-1}$ is a bounded linear operator. ■

Suppose that we are interested in the solution of the problem

$$Lu = (I + K)u = f \qquad \text{in } X \, .$$

The Fredholm alternative guarantees that the problem has a unique solution $u = L^{-1}f$ as long as $\lambda = 1$ is not an eigenvalue of K. If $\lambda = 1$ is an eigenvalue, by recalling that the adjoint of a compact operator is compact as well, the eigenspace $N(I+K^*)$ is of finite dimension and admits a basis $\{u_1^*, \dots, u_n^*\}$. For any linear and continuos operator $A : X \to X$ it is immediate to show that:

$$R(A)^\perp = N(A^*) \, , \tag{1.4.11}$$

where for a set $X_0 \subset X$ we define the annihilator $X_0^\perp = \{f \in X' \, : \, \langle f, x\rangle = 0 \;\, \forall x \in X_0\}$. The symbol $\langle \cdot \, , \cdot \rangle$ indicates the duality pairing between X and X'. Since $I+K$

has closed range, we can write $R(I+K) = N(I+K^*)_\perp$, where for a set $Y \subset X'$ we define $Y_\perp = \{x \in X \,:\, \langle f, x\rangle = 0 \;\forall f \in Y\}$. Then the problem has a solution if the load f satisfies the compatibility conditions $\langle u_i^*, f\rangle = 0$ for $i = 1, \ldots, n$.

To determine how the problem of interest fits into the general framework, we first define the trial space $W = U \times V$. Such a space is a complex Hilbert space endowed with the scalar product $(\cdot\,,\cdot) = (\cdot\,,\cdot)_U + (\cdot\,,\cdot)_V$, hence the theory we just presented simplifies in the sense that W' is identified to W and the duality pairing is replaced by a scalar product. By setting $\tilde{u}_D = (\tilde{\boldsymbol{u}}_D, \tilde{p}_D)$, problem (1.3.9) is equivalently restated as: *Find* $(\boldsymbol{u}, p) \in \tilde{u}_D + W$ *such that:*

$$\begin{aligned}
&\left[\int_{\Omega_e} E_{ijkl}u_{k,l}v_{i,j} + \int_{\Omega_a} \nabla p \cdot \nabla q\right] - \left[\int_{\Omega_e} \varrho_s \omega^2 u_i v_i\right.\\
&\quad \left. + \int_{\Omega_a} k^2 pq - i\omega \int_{\Gamma_{C,e}} \beta_{ij} u_j v_i - i\omega \int_{\Gamma_{C,a}} \varrho_f d pq - \int_{\Gamma_I} p\, v_i n_i \; + \omega^2 \varrho_f \int_{\Gamma_I} u_i n_i q\right]\\
&\qquad\qquad = \int_{\Omega_e} f_i v_i + \int_{\Gamma_{N,e}\cup\Gamma_{C,e}} g_i v_i + i\omega\varrho_f \int_{\Gamma_{N,a}\cup\Gamma_{C,a}} v_0 q \qquad \forall\, (\boldsymbol{v}, q) \in W\,.
\end{aligned}$$

Let $a(\cdot\,,\cdot) : W \times W \to \mathbb{C}$ be the term in the first pair of square brackets, $c(\cdot\,,\cdot) : W \times W \to \mathbb{C}$ be the opposite of the term in the second pair of square brackets and $l : W \to \mathbb{C}$ be the right-hand-side. Using a standard approach, we can absorb the finite energy lift $\tilde{u}_D$ into the right hand side and consider the problem: *Find* $u \in W$ *such that:*

$$b(u, w) = l(w) \qquad \forall\, w \in W\,.$$

where $u = (\boldsymbol{u}, p)$, $w = (\boldsymbol{v}, q)$, the bilinear form $b(\cdot\,,\cdot)$ is defined as $a(\cdot\,,\cdot) + c(\cdot\,,\cdot)$ and the right-hand-side is indeed $l(\cdot) - b(\tilde{u}_D, \cdot)$. This is done only for the sake of simplicity of the analysis. In practice, non homogeneous Dirichlet boundary conditions are dealt with in a different way, see Chapter 2.

The coercivity of form $a(\cdot\,,\cdot)$ is a consequence of Korn's and Poincaré's inequalities. Poincaré inequality states that there exists $C_P \geq 0$ such that $\|v\|_{0,\Omega} \leq C_P\|\nabla v\|_{0,\Omega}$ for all functions $v \in H^1(\Omega)$ with null trace on some set $\Gamma_0 \subset \partial\Omega$ with strictly positive (Hausdorff) measure. This subspace of $H^1(\Omega)$ is called $H^1_{\Gamma_0}(\Omega)$ and is trivially complete since it coincides with the kernel of the trace operator $\gamma v = v|_{\Gamma_0}$, which is linear and continuous as stated by the Trace Theorem. An immediate consequence of the Poincaré inequality is that $\|\nabla\cdot\|_{0,\Omega}$ is an equivalent norm on $H^1_{\Gamma_0}(\Omega)$, since $\|v\|^2_{1,\Omega} \leq (1 + C_P^2)\|\nabla v\|^2_{0,\Omega}$. Korn's inequality states that there exists $C_K \geq 0$ such that $\|\boldsymbol{v}\|^2_{1,\Omega} \leq C_K^2(\|\varepsilon(\boldsymbol{v})\|^2_{0,\Omega} + \|\boldsymbol{v}\|^2_{0,\Omega})$ for all functions $\boldsymbol{v} \in H^1(\Omega)^3$, see [31] for a proof. The domain Ω needs to satisfy some minimum regularity assumption, for example being a Lipschitz domain. A consequence of Korn's inequality is that $\|\varepsilon(\boldsymbol{v})\|_{0,\Omega}$ is an equivalent norm on the space $H^1_{\Gamma_0}(\Omega)^3$ whenever the (Hausdorff) measure of Γ_0 is strictly positive. We give a quick sketch of the proof. The first step is to show that all functions in $H^1(\Omega)^3$ satisfying $\|\varepsilon(\boldsymbol{v})\|_{0,\Omega} = 0$ are rigid body motions, hence $\boldsymbol{v} = 0$ by virtue of the null trace on Γ_0. This proves that $\|\varepsilon(\,\cdot\,)\|_{0,\Omega}$ is indeed *a* norm on $H^1_{\Gamma_0}(\Omega)^3$. To prove equivalence to $\|\cdot\|_{1,\Omega}$, we need to show that there exists a constant c_1, which is necessarily strictly positive, such that $\|\boldsymbol{v}\|_{1,\Omega} \leq c_1\|\varepsilon(\boldsymbol{v})\|_{0,\Omega}$. We argue by contradiction: for every n there exists $\boldsymbol{v}_n$ such that $\|\boldsymbol{v}_n\|_{1,\Omega} \geq n\|\varepsilon(\boldsymbol{v}_n)\|_{0,\Omega}$. Without loss of generality, we can assume $\|\boldsymbol{v}_n\|_{1,\Omega} = 1$, hence $\|\varepsilon(\boldsymbol{v}_n)\|_{0,\Omega} \to 0$. Moreover, by Rellich's theorem, there exists a subsequence, that we again call $\{\boldsymbol{v}_n\}$, which converges to some $\boldsymbol{v}_0$ in $L^2(\Omega)^3$. The Korn's inequality implies that $\|\boldsymbol{v}_0\|_{1,\Omega} \leq C_K \|\boldsymbol{v}_0\|_{0,\Omega}$, thus $\{\boldsymbol{v}_n\}$ is a Cauchy sequence in the complete space $H^1_{\Gamma_0}(\Omega)^3$ and therefore $\boldsymbol{v}_0$ belongs to $H^1_{\Gamma_0}(\Omega)^3$ as well. The continuity of the norm guarantees that $\|\varepsilon(\boldsymbol{v}_0)\|_{0,\Omega} = 0$ and the first step of our reasoning implies that $\boldsymbol{v}_0 = 0$. This contradicts the assumption that $\|\boldsymbol{v}_n\|_{1,\Omega} = 1$ for all n.

We have all of the tools to prove coercivity of form $a(\cdot\,,\cdot)$. Recalling the assumption $\lambda \geq 0$ on the first Lamé parameter, we have that:

$$\begin{aligned} &a(w,w) = \lambda \|\operatorname{div} \boldsymbol{v}\|^2_{0,\Omega_e} + 2\mu \|\varepsilon(\boldsymbol{v})\|^2_{0,\Omega_e} + \|\nabla q\|^2_{0,\Omega_a} \geq \\ &\geq 2\mu \tfrac{1}{c_1} \|\boldsymbol{v}\|^2_{1,\Omega_e} + \tfrac{1}{1+C_P^2} \|q\|^2_{1,\Omega_a} \geq \min\left\{2\mu \tfrac{1}{c_1}, \tfrac{1}{1+C_P^2}\right\} \left(\|\boldsymbol{v}\|^2_{1,\Omega_e} + \|q\|^2_{1,\Omega_a}\right) \geq \alpha \|w\|^2_W \,. \end{aligned}$$

Notice that $\alpha = \min\left\{2\mu/c_1, 1/(1{+}C_P^2)\right\}$ is a strictly positive constant since the shear modulus μ is strictly positive. This implies coercivity. Thus, operator $A : W \to W$ defined as $(Av, \overline{w}) = a(v,w)$ for all $v, w \in W$ is bounded below and selfadjoint as a consequence of the symmetry of $a(\cdot\,,\cdot)$. Since a bounded below operator is injective and has closed range, equation (1.4.11) implies that it is a bijection. Thus, since the open mapping theorem guarantees that A^{-1} is continuous as well, operator A is an isomorphism of W. Form $c(\cdot\,,\cdot)$ is *compact* by virtue of the compact embedding $H^1(\Omega) \overset{c}{\hookrightarrow} L^2(\Omega)$ that holds, i.e., every time that Ω is a bounded Lipschitz domain. We define a compact operator $C : W \to W$ such that $(Cv, \overline{w}) = b(v,w)$ for all $v, w \in W$. Finally, l is a continuous linear functional as soon as $f_i \in L^2(\Omega_e)$, $g_i \in L^2(\Gamma_e)$, and $v_0 \in L^2(\Gamma_a)$.

The generalized eigenvalues[1] of the problem of interest correspond to the resonant frequencies of the system. As a result of the dissipation of energy towards

---

[1] those values are called scattering frequencies by Lax and Phillips

infinity, the resonant frequencies pick up an imaginary component. Thus, since $\lambda = -1$ is not an eigenvalue of $K = A^{-1}C$, the Fredholm alternative implies that the bilinear form $b(\cdot\,,\cdot)$ satisfies the so called inf-sup condition, or Ladyzenskaya-Babuška-Brezzi condition

$$\exists\,\gamma > 0 \quad \text{such that} \qquad \sup_{\{w\,\in\,W\,:\,w\,\neq\,0\}} \frac{|b(v,w)|}{\|w\|} \geq \gamma\|v\| \quad \forall\, v \in W\ ,$$

where γ is called the inf-sup constant, or stability constant. Let $\{W_h\}$ be a family of finite dimensional subspaces of W that satisfies the approximation condition: for every $w \in W$ there exists a sequence $\{w_n\}$ where $w_n \in W_{h_n}$ such that $\lim_{n\to\infty} \|w - w_n\| = 0$. We consider the usual Bubnov-Galerkin approximation of the problem: *Find* $u_h \in W_h$ *such that:*

$$b(u_h, w_h) = l(w_h) \qquad \forall\, w_h \in W_h\ .$$

Contrary to the coercive case, the continuous inf-sup condition does not automatically imply a discrete inf-sup or, equivalently, discrete stability for the space W_h. Nevertheless, the following result guarantees that stability is recovered in the limit, as $h \to 0$, hence the problem is *asymptotically* stable.

Theorem 1.4.3 (Mikhlin)**.** *If the continuous problem has a unique solution, then there exists* $h_0 > 0$ *such that the discrete Ladyzenskaya-Babuška-Brezzi condition*

$$\exists\,\gamma_h > 0 \quad \textit{such that} \qquad \sup_{\{w_h\,\in\,W_h\,:\,w_h\,\neq\,0\}} \frac{|b(v_h,w_h)|}{\|w_h\|} \geq \gamma_h\|v_h\| \qquad \forall\, v_h \in W_h$$

holds for every $h < h_0$*.*

Proof. Let's assume, by contradiction, that for every $h_0 > 0$ there exists $h < h_0$ such that the discrete Ladyzenskaya-Babuška-Brezzi condition does not hold. Then we can construct a sequence $h_1 > \cdots > h_n > \cdots$ and a corresponding subfamily of spaces $\{W_{h_n}\}$ where, for every $\gamma_h > 0$, there exists $v_n \in W_{h_n}$ such that

$$\sup_{\{w_h \in W_{h_n} \,:\, w_h \neq 0\}} \frac{|b(v_n, w_h)|}{\|w_h\|} \leq \gamma_h \|v_n\| \,. \tag{1.4.12}$$

Without loss of generality, we assume $\gamma_h = 1/n$ and $\|v_n\| = 1$. Since $\{v_n\}$ is after all a bounded sequence in a complete space, we can extract a weakly convergent subsequence, that we call again $\{v_n\}$, such that $v_n \rightharpoonup v_0$. Such a convergence is indeed strong, as we show in the last part of the proof. Using the approximation condition for W, for every $w \in W$ there exists a sequence $\{w_k\}$, each element of which we can assume to belong to W_{h_n}—hence we set $k = n$—such that $w_n \to w$. By continuity of $b(\cdot\,,\cdot)$, we conclude that $b(v_0, w) = 0$ for all $w \in W$. Notice that $\|v_n\| \to \|v_0\|$, hence $\|v_0\| = 1$ and $v_0 \neq 0$, which contradicts the uniqueness of the solution. In order to prove strong convergence of $\{v_n\}$, we use the coercivity of $a(\cdot\,,\cdot)$:

$$\begin{aligned} \alpha\|v_0 - v_n\|^2 &\leq a(v_0 - v_n, v_0 - v_n) \\ &= b(v_0 - v_n, v_0 - v_n) - c(v_0 - v_n, v_0 - v_n) \\ &= b(v_0, v_0 - v_n) - b(v_n, v_0 - v_n) - c(v_0 - v_n, v_0 - v_n) \,. \end{aligned}$$

The three terms on the right-hand-side converge to zero, the first one because of of the weak convergence of $\{v_n\}$ and the last one because of the compactness of $c(\cdot\,,\cdot)$ and the fact that a weakly convergent sequence is bounded. In order to show convergence of the second term, we introduce a sequence $\{w_n\}$, where $w_n \in W_{h_n}$, such that $w_n \to v_0$. Then, by setting $M = \|B\|_*$, we conclude

$$\begin{aligned} |b(v_n, v_0 - v_n)| &\leq |b(v_n, v_0 - w_n)| + |b(v_n, w_n - v_n)| \\ &\leq M\|v_n\|\|v_0 - w_n\| + \frac{|b(v_n, w_n - v_n)|}{\|w_n - v_n\|}\|w_n - v_n\| \\ &\leq M\|v_0 - w_n\| + \tfrac{1}{n}(\|w_n\| + \|v_n\|) \to 0 \,, \end{aligned}$$

where we have used (1.4.12) and the fact that any convergent sequence is bounded.

■

The first consequence of the Mikhlin theorem is that for every h sufficiently small the discrete operator B_h, defined as the restriction of $A+C$ to W_h, is bounded below. This is sufficient to ensure existence and uniqueness of the solution, since the injectivity of B_h immediately implies surjectivity. Whenever the error $\|u - u_h\|$ is bounded by the best approximation error $\inf_{\{w_h \in W_h\}} \|u - w_h\|$ premultiplied by a mesh independent constant, the approximation is said to be *optimal*. In the case of $b(\cdot\,,\cdot)$ being coercive and symmetric, this constant is equal to one, whenever the energy norm $\|w\|_E = b(w,w)^{1/2}$ is considered in place of the usual norm $\|\cdot\|$ on W. In the general case, Babuška's theorem guarantees that if the discrete stability constant γ_h is independent of h, then the approximation is optimal. In the case of interest it can be proven, see L. Demkowicz, T. Oden [26], that

$$\|u - u_h\|_E \leq (1 + \mathrm{O}(h))\|u - w_h\|_E \qquad \forall w_h \in W_h \,,$$

where the energy norm is defined as $\|w\|_E = a(w,w)^{1/2}$. Since the constant $C_h = 1 + \mathrm{O}(h)$ is in the limit $h \to 0$ independent from h—and bounded!—we say that the problem is *asymptotically* optimal.

In practical applications it is crucial to assess the threshold value h_0. In general it is not possible to compute h_0 directly, however we can show its dependency upon the convergence of K_h, namely the restriction of K to W_h, to K. The inequality

$$\|(I+K)v_h\| \leq \|(I+K_h)v_h\| + \|K-K_h\|_*\|v_h\| \,,$$

where $\|\cdot\|_*$ is the operator norm associated to $\|\cdot\|$, implies that

$$\begin{aligned} \inf\nolimits_{\{v\,:\,\|v\|=1\}}\|(I+K)v\| \leq \inf\nolimits_{\{v_h\,:\,\|v_h\|=1\}}\|(I+K)v_h\| \leq \\ \leq \inf\nolimits_{\{v_h\,:\,\|v_h\|=1\}}\|(I+K_h)v_h\| + \|K-K_h\|_* \,. \end{aligned}$$

Recalling the definition of the inf-sup constant we have that

$$\begin{aligned} \gamma = \inf\nolimits_{\{v\,:\,\|v\|=1\}} \sup\nolimits_{\{w\,:\,\|w\|=1\}}|b(v,w)| = \\ = \inf\nolimits_{\{v\,:\,\|v\|=1\}} \sup\nolimits_{\{w\,:\,\|w\|=1\}}|(A(I+K)v,w)| = \inf\nolimits_{\{v\,:\,\|v\|=1\}}\|A\|_*\|(I+K)v\| \,, \end{aligned}$$

and similarly for the discrete inf-sup constant γ_h. Thus the previous inequality is equivalent to

$$\gamma_h \geq \gamma - \|A\|_*\|K-K_h\|_* \,.$$

This result shows that the region of asymptotic stability depends upon the continuous inf-sup constant and the rate of convergence of $\|K-K_h\|_*$ to zero. It also shows, as expectable, that the discrete inf-sup constant cannot be better the continuous one. In other words, we cannot expect a well-conditioned approximation of an ill-conditioned problem.

1.5 Perfectly Matched Layer

Many wave-propagation problems are naturally posed in unbounded domains. Since domain-based discretization methods such as finite elements can handle only bounded domains, a truncation of the unbounded physical domain to a bounded computational domain is necessary. This calls for the development of an *ad hoc* boundary condition to be imposed on the outer boundary, such that the original problem is not essentially altered, incoming non-physical waves are discarded, and outgoing physical waves are absorbed.

Among the various techniques, the Perfectly Matched Layer (PML) has achieved popularity due to its conceptual simplicity, ease of implementation and good performance. The original idea, due to Bèrenger [4] and subsequently developed by Chew and Weedon [17], is based on the analytic continuation of a real function into the complex plane. In fact, the original problem is modified in a way such that, outside of a truncating domain B_1 enclosing the scatterer Ω, the solutions of the problem become evanescent waves, with outgoing waves being exponentially decaying and incoming waves being exponentially growing as we move away from the origin. Thus, provided we choose a sufficiently large computational domain $\Omega_\infty \supset B_1$, we can discard the non-physical incoming waves by, *e.g.*, imposing a homogeneous Dirichlet condition on $\partial\Omega_\infty$. A brief literature review on the subject follows.

In [18], Collino and Monk derive a PML formulation for the Helmoltz equation in 2D, using polar coordinates. By explicitly constructing a solution in series of Hankel functions, they prove that the PML problem for a hard scatterer is well-posed for every wave number $k \in \mathbb{C}$, and its solution coincides with the original problem in

$B_1 \setminus \Omega$. When truncating the PML problem to a finite computational domain, they impose either a homogeneous Dirichlet condition or a homogeneous Robin condition of the outer boundary, arguing that the second one leads to a better numerical behavior. In either case, by means of the analytic Fredholm alternative, they prove that the problem is still well-posed for all real k's except a possibly empty discrete set. The analysis of the problem was extended by Lassas and Somersalo in [52]. Employing a double layer potential technique, they show that under certain assumptions on the absorption coefficients, the truncated PML problem is solvable for every k. Furthermore, they estimate the difference between solutions of the original problem and the truncated PML problem when the outer boundary is circular. They extend those 2D results to higher dimension and arbitrary smooth convex domains in [53]. In [11], Bramble and Pasciak, prove the existence and uniqueness of solution to the truncated electromagnetic PML problem in 3D, provided that the truncated domain is sufficiently large. The main difficulty is that techniques for acoustic problem do not carry over directly to the electromagnetic problem, since the electromagnetic operator has an infinite dimensional kernel, namely the gradients. Nevertheless, they start their discussion from the acoustic problem and employ a duality argument to show well-posedness of the truncated PML problem.

In the remainder of this section we borrow ideas from [11] to show that the PML problem is well posed. In fact, provided that a stretching function $r \mapsto \tilde{r} \in \mathbb{C}$ for the radial component is suitably chosen, we can split the variational formulation into a coercive contribution and a compact contribution and apply the Fredholm alternative.

Let (r,θ,φ) be the usual spherical coordinates $x = r\cos\theta\sin\varphi$, $y = r\sin\theta\sin\varphi$, $z = r\cos\varphi$. In oder to obtain the Bèrenger problem for the Helmholtz equation, we need to formally replace r by a complex-valued radial component $\tilde{r}$. Following [18] and [52], let $B_i = B_{r_i}(0)$, choose radii r_1, r_2 such that $\Omega \subset B_1 \subset B_2$, and construct a function $\tilde{\sigma} \in C^2(\mathbb{R}^+;\mathbb{R})$ such that

$$\tilde{\sigma}(r) = \begin{cases} 0 & 0 \le r \le r_1 \\ \text{increasing} & r_1 \le r \le r_2 \\ \sigma_0 & r \ge r_2 \end{cases} ,$$

where σ_0 is a positive constant. Let us further define $\tilde{d} = 1 - i\tilde{\sigma}$, $\sigma = \tilde{\sigma} + r\tilde{\sigma}'$ and $d = 1 - i\sigma$. Since $\tilde{\sigma}' \ge 0$, the definition of σ implies that $\sigma \ge \tilde{\sigma}$. The stretched radial coordinate $\tilde{r}$ and its derivative $\tilde{r}'$ are defined as

$$\tilde{r} = \tilde{d}r \qquad ; \qquad \tilde{r}' = 1 - i(\tilde{\sigma} + \varrho\tilde{\sigma}') = 1 - i\sigma = d\,.$$

The complex extension implies that the differential operators need to be modified as well, by substituting derivatives with respect to r by derivatives with respect to $\tilde{r}$. A way to proceed is to define the stretched gradient, divergence and Laplacian:

$$\begin{aligned} \tilde{\nabla} f &= \mathbf{e}_r \frac{1}{d}\frac{\partial f}{\partial r} + \mathbf{e}_\varphi \frac{1}{r\tilde{d}}\frac{\partial f}{\partial \varphi} + \mathbf{e}_\theta \frac{1}{r\tilde{d}\sin\varphi}\frac{\partial f}{\partial \theta}\,, \\ \tilde{\operatorname{div}} \boldsymbol{f} &= \frac{1}{\tilde{d}^2 r^2 d}\frac{\partial}{\partial r}(\tilde{d}^2 r^2 f_r) + \frac{1}{\tilde{d} r\sin\varphi}\frac{\partial}{\partial\varphi}(\sin\varphi f_\varphi) + \frac{1}{\tilde{d} r\sin\varphi}\frac{\partial f_\theta}{\partial\theta}\,, \\ \tilde{\Delta} f &= \frac{1}{r^2\tilde{d}^2 d}\frac{\partial}{\partial r}\left(\frac{r^2\tilde{d}^2}{d}\frac{\partial f}{\partial r}\right) + \frac{1}{r^2\tilde{d}^2\sin\varphi}\frac{\partial}{\partial\varphi}\left(\sin\varphi\frac{\partial f}{\partial\varphi}\right) + \frac{1}{r^2\tilde{d}^2\sin^2\varphi}\frac{\partial^2 f}{\partial\theta^2}\,. \end{aligned}$$

Indeed, through direct computation, we can *verify* that $\tilde{\Delta} f = \tilde{\operatorname{div}}\big(\tilde{\nabla} f\big) = 1/d\tilde{d}\operatorname{div}(\Lambda\nabla f)$, where $\Lambda = \operatorname{diag}(\tilde{d}/d, d/\tilde{d}, d/\tilde{d})$. Most authors replace Δ by $\tilde{\Delta}$ in the Helmholtz equation and seek a solution $p = p(\tilde{r},\theta,\varphi)$ that satisfies the so called Bèrenger equation

$$-\frac{1}{d\tilde{d}}\operatorname{div}(\Lambda\nabla p) - k^2 p = 0 \qquad \text{in } \Omega^c\,.$$

The Sommerfeld condition of the Helmholtz equation, $\lim_{r\to\infty}(\nabla p \cdot \boldsymbol{r} + rikp) = 0$, where $\boldsymbol{r}$ is the position vector, translates into requiring that $|p|$ is uniformly bounded as $r \to \infty$. The main point that makes the Bèrenger equation attractive is that $\Lambda = I$ for $r \geq r_2$, hence it reduces to the Helmholtz equation with a complex wave number in $\mathbb{R}^3 \setminus \overline{B}_2$. Once a boundary condition is set on Γ, existence and uniqueness for the strong problem

$$-\operatorname{div}(\Lambda \nabla p) - k^2 d\tilde{d} p = 0 \qquad \text{in } \Omega^c\,, \tag{1.5.13a}$$

$$p = p_0 \qquad \text{on } \Gamma\,, \tag{1.5.13b}$$

$$|p| \text{ uniformly bounded} \qquad \text{at } \infty\,, \tag{1.5.13c}$$

can be shown through a direct computation as Collino and Monk do in [18].

While problem (1.5.13) gives a good insight into the physical meaning of the PML, we believe it is not the right starting point to find a variational formulation. Indeed, if we were to develop a variational formulation of (1.5.13) using standard techniques, as done in [11], we would lose symmetry, due to the Jacobian of the transformation from cartesian to spherical coordinates that would appear under the integral sign. Although this is not *a priori* a problem, it involves a number of difficulties that can be overcome using general systems of curvilinear coordinates. Let $u_j = u_j(x_i)$ be an arbitrary system of curvilinear coordinates, then differential operators are defined as

$$\nabla f = \frac{\partial f}{\partial u_j}\boldsymbol{a}^j \qquad ; \qquad \operatorname{div} \boldsymbol{f} = \frac{\partial \boldsymbol{f}}{\partial u_j} \cdot \boldsymbol{a}^j\,,$$

where $\boldsymbol{a}^j = \partial u_j / \partial x_l\, \mathbf{e}_l$ are the cobasis vectors. We can use standard integration by parts in cartesian coordinates to verify that the above definition is indeed correct and leads to the desired integration by parts formula in curvilinear coordinates:

$$\begin{aligned}\int_\Omega \frac{\partial f}{\partial u_j}\boldsymbol{a}^j \cdot g_i\mathbf{e}_i = \int_\Omega \frac{\partial f}{\partial u_j}\frac{\partial u_j}{\partial x_l}\mathbf{e}_l \cdot g_i\mathbf{e}_i = \int_\Omega \frac{\partial f}{\partial x_l} g_i(\mathbf{e}_l \cdot \mathbf{e}_i) = \int_\Omega \frac{\partial f}{\partial x_i} g_i = \\ = \int_\Gamma f n_i g_i - \int_\Omega f\frac{\partial g_i}{\partial x_i} = \int_\Gamma f n_i g_i - \int_\Omega f \frac{\partial g_i}{\partial x_k}\delta_{kl}\delta_{il} = \\ = \int_\Gamma f n_i g_i - \int_\Omega f \frac{\partial g_i}{\partial x_k}\frac{\partial x_k}{\partial u_j}\frac{\partial u_j}{\partial x_l}\,\mathbf{e}_i\cdot\mathbf{e}_l = \int_\Gamma f n_i g_i - \int_\Omega f\frac{\partial g_i}{\partial u_j}\mathbf{e}_i \cdot \frac{\partial u_j}{\partial x_l}\mathbf{e}_l = \\ = \int_\Gamma f n_i g_i - \int_\Omega f \frac{\partial \boldsymbol{g}}{\partial u_j}\cdot \boldsymbol{a}^j \,.\end{aligned}$$

In the above equalities n_i is the i-th component of the outward normal, Ω is an arbitrary domain, and $\Gamma = \partial\Omega$. The fundamental observation is that the stretched differential operators defined above are the usual operators, according to the curvilinear system of coordinates $(\tilde{r}, \theta, \varphi)$. This allows us to use the usual vector calculus identities, without any special treatment of $\tilde{\nabla}$ and $\tilde{\operatorname{div}}$. Indeed, the cobasis vectors are

$$\boldsymbol{a}^{\tilde{r}} = \mathbf{e}_r \qquad ; \qquad \boldsymbol{a}^\theta = \frac{1}{\tilde{r}}\mathbf{e}_\theta \qquad ; \qquad \boldsymbol{a}^\varphi = \frac{1}{\tilde{r}\sin\varphi}\mathbf{e}_\varphi \,,$$

and it is trivial to verify that

$$\tilde{\nabla} f = \boldsymbol{a}^{\tilde{r}}\frac{\partial f}{\partial \tilde{r}} + \boldsymbol{a}^\theta \frac{\partial f}{\partial \theta} + \boldsymbol{a}^\varphi \frac{\partial f}{\partial \varphi} \,.$$

Similar computations, yet more tedious, hold in the case of the divergence operator.

Since we established that $\tilde{\Delta} f = \tilde{\operatorname{div}}(\tilde{\nabla} f)$, it is perfectly equivalent (at least up to regularity assumptions) to work on the the second order equation or the first order system. Moreover, the variational formulation of the stretched problem is exactly the form $b_{aa}(\cdot\,,\cdot)$ defined in (1.3.10d), provided that the gradient and the infinitesimal volume element are computed according to the system $(\tilde{r}, \theta, \varphi)$. In practice we have that:

$$\tilde{b}(p,q) = \int_{\Omega^c} \big(\tilde{\nabla} p \cdot \tilde{\nabla} q - k^2 pq\big) d\tilde{V} \,. \tag{1.5.14}$$

where the test function q belongs to $H^1_0(\Omega^c)$. This choice corresponds to the case of a Dirichlet boundary condition on the scatterer. Treatment of the Neumann problem is done in a similar fashion. The above formulation is to be understood with respect to the $(\tilde{r}, \theta, \varphi)$ system of coordinates. By noticing that $\tilde{d}\,\tilde{\nabla} q = (\tilde{d}/d)\,\nabla_r q + \nabla_S q$, where ∇_r and ∇_S are the radial and spherical components of the gradient respectively, and $d\tilde{V} = d\tilde{d}^2\, dV$, the above variational formulation can be expressed through an integral in standard spherical coordinates:

$$\tilde{b}(p,q) = \int_{\Omega^c} \left(\frac{\tilde{d}}{d} \,\nabla_r p \cdot \nabla_r q + d\,\nabla_S p \cdot \nabla_S q - k^2 d\tilde{d}^2 pq \right) dV \,.$$

In order to show that the stretched problem is well posed, we show now that there exists a decomposition $\tilde{b}(\cdot\,,\cdot) = a(\cdot\,,\cdot) + c(\cdot\,,\cdot)$ such that $c(\cdot\,,\cdot) : H^1_0(\Omega^c) \times H^1_0(\Omega^c) \to \mathbb{C}$ is compact and $a(\cdot\,,\cdot) : H^1_0(\Omega^c) \times H^1_0(\Omega^c) \to \mathbb{C}$ satisfies the coercivity condition:

$$\text{there exists } \alpha > 0 \text{ such that} \quad \Re\, a(q, \overline{q}) \geq \alpha \|q\|^2_{1,\Omega^c} \quad \forall q \in H^1_0(\Omega^c)\,. \tag{1.5.15}$$

We immediately obtain that:

$$\begin{aligned}
\tilde{b}(p,q) &= \int_{\Omega^c} \big(d\tilde{d}^2 \tilde{\nabla} p \cdot \tilde{\nabla} q - k^2 d\tilde{d}^2 pq \big) = \\
&= \int_{\Omega^c} \big(d\tilde{d}^2 \tilde{\nabla} p \cdot \tilde{\nabla} q - k^2 d(\tilde{d}^2 - d_0^2 + d_0^2) pq \big) = \\
&= \int_{\Omega^c} \big(d\tilde{d}^2 \tilde{\nabla} p \cdot \tilde{\nabla} q - k^2 d d_0^2 pq \big) + \int_{\Omega^c} k^2 d(\tilde{d}^2 - d_0^2) pq \,,
\end{aligned}$$

where we have set $d_0 = 1 - i\sigma_0$. Let us define

$$a(p,q) = \int_{\Omega^c} \big(d\tilde{d}^2\tilde{\nabla}p \cdot \tilde{\nabla}q - k^2 d d_0^2 pq\big) \qquad ; \qquad c(p,q) = \int_{\Omega^c} k^2 d\big(\tilde{d}^2 - d_0^2\big)pq\,.$$

We can now prove the following theorem.

Theorem 1.5.1. *Let $\Phi \in L^2(\Omega^c)$ be an arbitrary load. The problem:*

$$\textit{Find } p \in H_0^1(\Omega^c) \textit{ such that } \tilde{b}(p,q) = (\Phi, q)_{\Omega^c} \textit{ for all } q \in H_0^1(\Omega^c) \tag{1.5.16}$$

is well posed and the estimate $\|p\|_{1,\Omega^c} \le C\|\Phi\|_{0,\Omega^c}$ holds.

Proof. Let us show that $c(\cdot\,,\cdot)$ is compact. Let $\Omega_2 = B_2\backslash\overline{\Omega}$. Since $\operatorname{supp}\big(\tilde{d}^2 - d_0^2\big) \subset \Omega_2$, then the domain of integration Ω^c can be replaced by the bounded domain Ω_2; let $R : H_0^1(\Omega^c) \to H^1(\Omega_2)$ be the restriction on Ω_2 and $J : H^1(\Omega_2) \hookrightarrow L^2(\Omega_2)$ the embedding, which is a compact operator. If we define $\widetilde{K} : L^2(\Omega_2) \to H_0^1(\Omega^c)'$ as $\widetilde{K}p = k^2 d\big(\tilde{d}^2 - d_0^2\big)p$, then we have that $c(p,q) = \langle \widetilde{K}JRp, q\rangle$ which shows that $c(\cdot\,,\cdot)$ is a compact form. Finally let us show that $a(\cdot\,,\cdot)$ satisfies condition (1.5.15). Recalling the relationship $\tilde{d}\,\tilde{\nabla} = (\tilde{d}/d)\,\nabla_r + \nabla_S$, we immediately have that

$$\begin{aligned} -\,a(q,\overline{q}) &= \int_{\Omega^c} \left(k^2 d d_0^2 q\overline{q} - (\tilde{d}^2/d)\nabla_r q \cdot \overline{\nabla_r q} - d\,\nabla_S q \cdot \overline{\nabla_S q}\right) = \\ &= \int_{\Omega^c} \left(k^2 d d_0^2 |q|^2 - (\tilde{d}^2/d)|\nabla_r q|^2 - d|\nabla_S q|^2\right), \end{aligned}$$

hence

$$-\Re\, a(q,\overline{q}) = \int_{\Omega^c} \left(k^2\, \Re(d d_0^2)|q|^2 - \Re(\tilde{d}^2/d)|\nabla_r q|^2 - |\nabla_S q|^2\right).$$

Let us estimate $\Re(dd_0^2)$ and $\Re(\tilde{d}^2/d)$ in order to assess the sign of $\Re\, a(q,\overline{q})$. Recalling that $\sigma \geq \tilde{\sigma}$ and assuming $\sigma_0 > 1$, we have that

$$\begin{aligned}
\Re(dd_0^2) &= 1 - \sigma_0^2 - 2\sigma_0\sigma \leq 1 - \sigma_0^2 \leq -C_1 < 0\,,\\
\Re(\tilde{d}^2/d) &= \frac{1-\tilde{\sigma}^2+2\tilde{\sigma}\sigma}{1+\sigma^2} \geq \frac{1-\tilde{\sigma}^2+2\tilde{\sigma}^2}{1+\sigma^2} = \frac{1+\tilde{\sigma}^2}{1+\sigma^2} \geq C_2 > 0\,.
\end{aligned}$$

Therefore

$$\begin{aligned}
-\Re\, a(q,\overline{q}) &= \int_{\Omega^c} \Big(\underbrace{k^2\,\Re(dd_0^2)|q|^2}_{\leq 0} - \underbrace{\Re(\tilde{d}^2/d)\,|\nabla_r q|^2}_{\geq 0} - \underbrace{|\nabla_S q|^2}_{\geq 0} \Big)\\
&\leq \int_{\Omega^c} \big(-C_1 k^2 |q|^2 - C_2 |\nabla_r q|^2 - |\nabla_S q|^2 \big)\\
&\leq -\alpha \|q\|^2_{1,\Omega^c}\,.
\end{aligned}$$

Hence we have that $\Re\, a(q,\overline{q}) \geq \alpha\|q\|^2_{1,\Omega^c}$ as desired. Uniqueness of the solution of the strong problem (1.5.13) was proved by Collino and Monk in [18], through a direct computation. By the Fredholm alternative, uniqueness of the solution implies existence. The estimate is a very well known consequence of the generalized Lax-Milgram Lemma, see, e.g., [62], page 518. ■

As a final step let us introduce the truncated PML problem. Let $\Omega_\infty = B_R \backslash \overline{\Omega}$, where B_R is the ball of radius $R > r_2$ centered at the origin. The truncated problem, whose solution we denote by p_∞, is

$$-\tilde{\Delta} p_\infty - k^2 p_\infty = 0 \qquad \text{in } \Omega_\infty\,, \tag{1.5.17a}$$

$$p_\infty = p_0 \qquad \text{on } \Gamma\,, \tag{1.5.17b}$$

$$p_\infty = 0 \qquad \text{on } \Gamma_\infty := \partial B_R\,. \tag{1.5.17c}$$

It is immediate to show that the variational formulation for the truncated problem is:

$$\text{Find } p_\infty \in H_0^1(\Omega_\infty) \text{ such that } \tilde{b}_\infty(p_\infty, q) = 0 \text{ for all } q \in H_0^1(\Omega_\infty)\,,$$

where

$$\begin{aligned}\tilde{b}_\infty(p_\infty, q) &= \int_{\Omega_\infty} \left(d\tilde{d}^2 \tilde{\nabla} p_\infty \cdot \tilde{\nabla} q - k^2 d\tilde{d}^2 p_\infty q\right) \\ &= \int_{\Omega_\infty} \left(d\tilde{d}^2 \tilde{\nabla} p_\infty \cdot \tilde{\nabla} q - k^2 d d_0^2 p_\infty q\right) + \int_{\Omega_\infty} k^2 d(\tilde{d}^2 - d_0^2) p_\infty q \\ &= a_\infty(p_\infty, q) + c_\infty(p_\infty, q)\,.\end{aligned}$$

Since $a_\infty(p_\infty, q) = a(\tilde{p}_\infty, \tilde{q})$ and $c_\infty(p_\infty, q) = c(\tilde{p}_\infty, \tilde{q})$, where $\tilde{\cdot}$ is the trivial extension to Ω^c, then $a_\infty(\cdot\,,\cdot)$ and $c_\infty(\cdot\,,\cdot)$ are, respectively, a coercive and a compact form over the reduced space $H_0^1(\Omega_\infty) \times H_0^1(\Omega_\infty)$. As in the case of the problem posed in the unbounded domain Ω^c, the Fredholm alternative guarantees that uniqueness implies existence of the solution. In order to prove uniqueness, we employ the technique used by Bramble and Pasciak in [11]. We require the following two lemmas that are taken directly from [11]. The first one is an internal estimate for the solution of the Helmholtz equation.

Lemma 1.5.2. *Let $u \in H^1(D)$ satisfy the Helmholtz equation, with a possibly complex wave number k, in the domain D. Then there exists a positive constant C that depends on k such that $\|u\|_{2,D_1} \leq C\|u\|_{0,D}$, where D_1 is any domain whose closure is contained in D.*

Proof. Let us introduce intermediate domains $\overline{D}_1 \subset D_2$, $\overline{D}_2 \subset D_3 := D$ and define smooth cut-off functions ρ_i such that $\rho_i = 1$ on D_i and $\rho_i = 0$ on D_{i+1}^c for $i = 1, 2$. Let

$u_i = \rho_i u$, so that u_i and its gradient vanish on ∂D_i. For every function $\phi \in H^1(D)$, using integration by parts twice, we have that:

$$\begin{aligned}\int_D (\nabla u_2 \cdot \nabla\phi + u_2\phi) &= \int_D (-\Delta u_2 + u_2)\phi = \\ &= \int_D (-\Delta\rho_2 u - 2\nabla\rho_2 \cdot \nabla u - \rho_2 \Delta u + \rho_2 u)\phi = \\ &= \int_D (-\Delta\rho_2 u\phi + 2\,\mathrm{div}(\nabla\rho_2\phi)u + k^2\rho_2 u\phi + \rho_2 u\phi) = \\ &= \int_D (2\nabla\rho_2 u \cdot \nabla\phi + (\Delta\rho_2 + k^2\rho_2 + \rho_2)u\phi) =: F_2(\phi)\,.\end{aligned}$$

The functional F_2 is defined on $H^1(D)$ and the Cauchy-Swartz inequality implies that $\|F_2\|_* \leq C\|u\|_{0,D}$. Since $u = u_2$ in D_2, we have that:

$$\|u\|_{1,D_2} = \|u_2\|_{1,D_2} \leq \|u_2\|_{1,D} \leq C\|u\|_{0,D}\,. \tag{1.5.18}$$

Now, for every $\phi \in H^1(\mathbb{R}^3)$, using integration by parts once so that the ∇u term is retained, we find that

$$\begin{aligned}\int_{\mathbb{R}^3} (\nabla u_1 \cdot \nabla\phi + u_1\phi) &= \int_{D_2} (\nabla u_1 \cdot \nabla\phi + u_1\phi) = \int_{D_2} (-\Delta u_1 + u_1)\phi = \\ &= \int_{D_2} (-\Delta\rho_1 u - 2\nabla\rho_1 \cdot \nabla u - \rho_1\Delta u + \rho_1 u)\phi = \\ &= \int_{\mathbb{R}^3} (-\Delta\rho_1 u - 2\nabla\rho_1 \cdot \nabla u + k^2\rho_1 u + \rho_1 u)\phi\,.\end{aligned}$$

Since we assumed $u \in H^1(D)$, the function $f_1 = (-\Delta\rho_1 u - 2\nabla\rho_1 \cdot \nabla u + k^2\rho_1 u + \rho_1 u)$ belongs to $L^2(\mathbb{R}^3)$. The function u_1 solves the problem:

$$-\Delta u_1 + u_1 = f_1 \qquad \text{a.e. in } \mathbb{R}^3\,.$$

The Fourier characterization of $H^2(\mathbb{R}^3)$ via the Bessel potential implies that u_1 belongs to $H^2(\mathbb{R})^3$ and $\|u_1\|_{2,\mathbb{R}^3} = \|f_1\|_{0,\mathbb{R}^3}$. Moreover, using the fact that $\operatorname{supp} f_1 \subset \overline{D}_2$

and the Cauchy-Swartz inequality we immediately have that $\|f_1\|_{0,\mathbb{R}^3} = \|f_1\|_{0,D_2} \leq C\|u\|_{1,D_2}$. Finally, recalling that $u = u_1$ in D_1, and using the relation (1.5.18), we obtain

$$\|u\|_{2,D_1} = \|u_1\|_{2,D_1} \leq \|u_1\|_{2,D_2} \leq C\|u\|_{1,D_2} \leq C\|u\|_{0,D} \, ,$$

which is the desired result. ■

The constant C that enters the estimate provided by Lemma 1.5.2 depends upon the wave number, but is independent of the size of the subdomain D_1. This is consistent with the fact that D_1 could be unbounded. It is crucial to notice that the estimate provided by Lemma 1.5.2 is only internal, since the closure of subdomain D_1 has to be contained in D. This implies that $\partial p_\infty/\partial n = \nabla p_\infty \cdot \mathbf{n}$ belongs to $H^{-1/2}(\partial\Omega_\infty)$. Moreover, the trace of $\partial p_\infty/\partial n$ on Γ_∞, which is only a portion of the boundary, is an element of the dual of the space $H_{00}^{1/2}(\Gamma_\infty) = \{v \in H^{1/2}(\Gamma_\infty) \, : \, \tilde{v} \in H^{1/2}(\partial\Omega_\infty)\}$, where $\tilde{v}$ is the trivial extension over the whole boundary $\partial\Omega_\infty$. The norm on the space $H^{1/2}(\Gamma_\infty)$ is defined as

$$\|v\|_{1/2,\Gamma_\infty} = \inf\{\|w\|_{1,\Omega_\infty} \, : \, w \in H^1(\Omega_\infty) \, , \, w|_{\Gamma_\infty} = v\} \, ,$$

and the norm on $H_{00}^{1/2}(\Gamma_\infty)$ is defined accordingly. The norm on the dual space is defined through the duality pairing in the usual way:

$$\|v\|_{H_{00}^{1/2}(\Gamma_\infty)'} = \sup_{\phi \in H_{00}^{1/2}(\Gamma_\infty)} \frac{\left|{}_{H_{00}^{1/2}(\Gamma_\infty)'}\langle v, \phi\rangle_{H_{00}^{1/2}(\Gamma_\infty)}\right|}{\|\phi\|_{1/2,\Gamma_\infty}} = \sup_{\phi \in H_{00}^{1/2}(\Gamma_\infty)} \frac{|\langle v, \tilde{\phi}\rangle_{\partial\Omega_\infty}|}{\|\phi\|_{1/2,\Gamma_\infty}} \, ,$$

where $\langle \cdot \, , \cdot \rangle_{\partial\Omega_\infty}$ is the duality pairing between $H^{-1/2}(\partial\Omega_\infty)$ and $H^{1/2}(\partial\Omega_\infty)$. In the case of interest things are simpler since $\partial\Omega_\infty = \Gamma_\infty \cup \Gamma$ and Γ_∞ and Γ are disjoint.

This implies that $H^{1/2}_{00}(\Gamma_\infty) = H^{1/2}(\Gamma_\infty)$, hence $\partial p_\infty/\partial n$ belongs to $H^{-1/2}(\Gamma_\infty)$ and its norm can be computed as

$$\left\| \frac{\partial p_\infty}{\partial n} \right\|_{-1/2,\Gamma_\infty} = \sup_{\phi \in H^{1/2}(\Gamma_\infty)} \frac{\left| \left\langle \dfrac{\partial p_\infty}{\partial n}, \phi \right\rangle_{\Gamma_\infty} \right|}{\|\phi\|_{1/2,\Gamma_\infty}} \ .$$

The technicality of evaluating $\|\partial p_\infty/\partial_n\|_{-1/2,\Gamma_\infty}$ will show up in Theorem 1.5.4.

We state the second lemma in the context of our problem. To do so, we introduce some notation. Let Ω_3 and Ω_4 be domains contained in Ω_∞ and defined as $\Omega_i = B_{r_i} \setminus \overline{\Omega}$ for $r_4 \geq r_3 \geq r_2$. We denote the outer boundary of domain Ω_i by Γ_i. The situation in illustrated in Figure 1.1.

Lemma 1.5.3. *Define* $S_\varepsilon = \{x \in \mathbb{R}^3 \ : \ \operatorname{dist}(x, \Gamma_\infty) < \varepsilon\}$. *Let u be bounded at infinity and satisfy the Helmholtz equation with wave number* $kd_0 = k(1 + i\sigma_0)$, *possibly complex, on the domain* $\Omega^c \setminus \overline{\Omega}_2$. *Then the estimate*

$$\|u\|_{0,S_\varepsilon} \leq Ce^{-\sigma_0 kR} \|u\|_{0,\Omega_4}$$

holds for every $R > r_4$ *and* ε *sufficiently small so that* $\overline{S}_\varepsilon \subset \Omega^c \setminus \overline{\Omega}_4$.

The proof of Lemma 1.5.3 can be found in [11]. It relies on the use of the fundamental solution for the Helmholtz equation, as well as Lemma 1.5.2. The message of Lemma 1.5.3 is that we can control $\|p\|_0$ in a neighborhood of Γ_∞ in terms of its norm on the set Ω_4. We have all of the tools to prove uniqueness of the solution p_∞ to the truncated problem.

Theorem 1.5.4. *Let* $p_\infty \in H^1_0(\Omega_\infty)$ *satisfy the strong problem (1.5.17) with* $p_0 = 0$. *Then, for R chosen large enough,* $p_\infty = 0$.

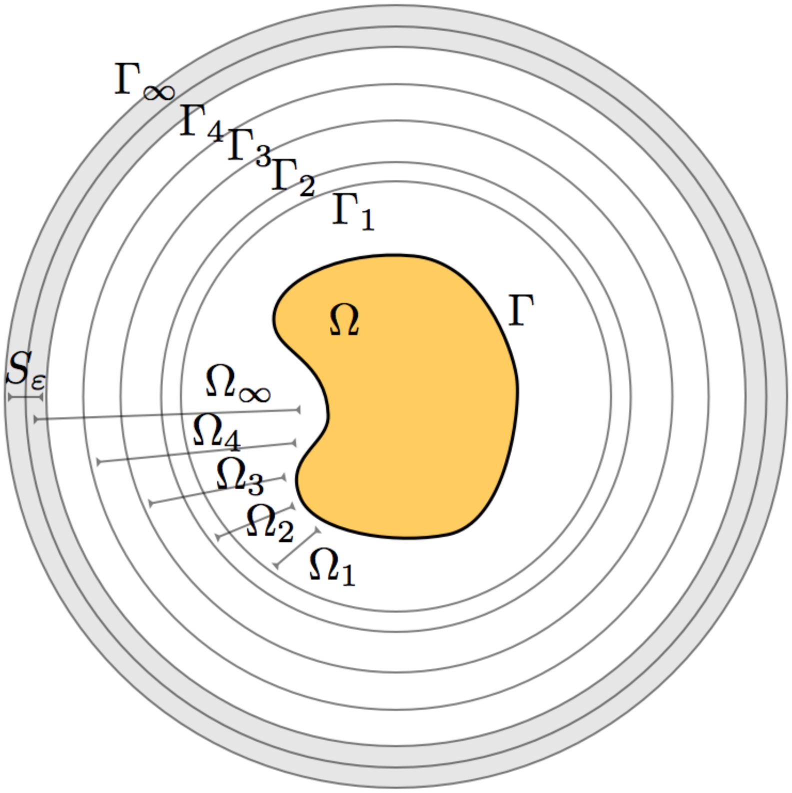


Figure 1.1: PML layer and truncating domains.

Proof. The proof relies on the concept of analytic continuation. Since the solution p_∞ is analytic inside Ω_∞, as implied by general theory of partial differential equations, it is sufficient to show that it vanishes on Ω_4 to conclude that it vanishes on the entire domain Ω_∞. In order to evaluate $\|p_\infty\|_{0,\Omega_4}$, we introduce the auxiliary problem:

$$\text{Find } p_\Phi \in H_0^1(\Omega^c) \text{ such that } \tilde{b}(p_\Phi, q) = (\Phi, q) \text{ for all } q \in H_0^1(\Omega^c)\,, \qquad (1.5.19)$$

where $\Phi \in L^2(\Omega^c)$ is a load such that $\operatorname{supp}\Phi \subset \Omega_4$. Since (the trivial extension of) p_∞ belongs to $H_0^1(\Omega^c)$, we can select $q = p_\infty$ and, recalling that p_∞ satisfies (1.5.13a) on Ω_∞, using integration by parts over Ω_∞, we immediately obtain that

$$(\Phi, p_\infty) = \tilde{b}(p_\Phi, p_\infty) = \Big\langle \frac{\partial p_\infty}{\partial n}, p_\Phi \Big\rangle_{\Gamma_\infty} = \Big\langle \frac{\partial p_\infty}{\partial n}, \rho p_\Phi \Big\rangle_{\Gamma_\infty} ,$$

where in the last equality we inserted a cut-off function ρ which is equal to one on Γ_∞ and such that $\operatorname{supp} \rho = \overline{D}_1 \subset S_\varepsilon$. By doing so, we are able to *localize* p_Φ so that we can use Lemma 1.5.3, Lemma 1.5.2 and Theorem 1.5.1 to obtain the following estimate:

$$\|\rho p_\Phi\|_{1/2,\Gamma_\infty} \le C\|p_\Phi\|_{1,D_1} \le C\|p_\Phi\|_{0,S_\varepsilon} \le Ce^{-\sigma_0 kR}\|p_\Phi\|_{0,\Omega_4} \le Ce^{-\sigma_0 kR}\|\Phi\|_{0,\Omega_4} .$$

Hence we obtain that

$$|(\Phi, p_\infty)| = \Big|\Big\langle \frac{\partial p_\infty}{\partial n}, p_\Phi \Big\rangle_{\Gamma_\infty}\Big| \le C\, e^{-\sigma_0 kR}\|\Phi\|_{0,\Omega_4} \Big\|\frac{\partial p_\infty}{\partial n}\Big\|_{-1/2,\Gamma_\infty} , \tag{1.5.20}$$

and we are left with the task of estimating the fractional norm in terms of $\|p_\infty\|_{0,\Omega_4}$. Notice that p_∞ solves the Helmholtz with the complex wave number $k(1 + i\sigma_0)$ in $\Omega_\infty \setminus \overline{\Omega}_2$. The idea is to use Green's second identity to write the duality pairing $\langle \partial p_\infty/\partial n, \phi\rangle_{\Gamma_\infty}$, for an arbitrary test function ϕ, in terms of duality pairings on the intermediate boundary Γ_3, so that we can employ Lemma 1.5.2 with a subdomain D_1 containing Γ_3. This is achieved through an additional auxiliary problem. For every ϕ belonging to $H^1_{\Gamma_2}(\Omega_\infty \setminus \overline{\Omega}_2)$, we consider the problem:

Find $\psi \in H^1_0(\Omega_\infty \setminus \overline{\Omega}_2)$ such that:

$(\nabla\psi, \nabla q) - k^2 d_0^2\, (\psi, q) = (\nabla\phi, \nabla q) - k^2 d_0^2\, (\phi, q)$ for all $q \in H^1_0(\Omega_\infty \setminus \overline{\Omega}_2)$.

This problem is well posed since wave number $kd_0 = k(1 + i\sigma_0)$ has a non zero imaginary part. The solution is $\psi = \phi$ in $\Omega_\infty \setminus \overline{\Omega}_2$ and the following estimate holds:

$$\|\psi\|_{1,\Omega_\infty \setminus \overline{\Omega}_2} \le C\|\phi\|_{1,\Omega_\infty \setminus \overline{\Omega}_2} . \tag{1.5.21}$$

We set $h = \phi - \psi$, notice that $h = \phi$ on Γ_∞, and use Green's second identity:

$$0 = \int_{\Omega_\infty \setminus \overline{\Omega}_3} (p_\infty \, \Delta h - h \, \Delta p_\infty) = \Big\langle \frac{\partial h}{\partial n}, p_\infty \Big\rangle_{\Gamma_\infty \cup \Gamma_3} - \Big\langle \frac{\partial p_\infty}{\partial n}, h \Big\rangle_{\Gamma_\infty \cup \Gamma_3} =$$

$$= \Big\langle \frac{\partial h}{\partial n}, p_\infty \Big\rangle_{\Gamma_3} - \Big\langle \frac{\partial p_\infty}{\partial n}, \phi \Big\rangle_{\Gamma_\infty} - \Big\langle \frac{\partial p_\infty}{\partial n}, h \Big\rangle_{\Gamma_3} .$$

In the above identity the duality pairings over Γ_3 can indeed be interpreted as scalar products, since both $\partial p_\infty / \partial n$ and $\partial h / \partial n$ belong to H^2 in a neighborhood of Γ_3. Employing a standard trace inequality and Lemma 1.5.2, with D_1 a domain containing Γ_3 and such that $\overline{D}_1 \subset \Omega_4 \setminus \overline{\Omega}_2$, we have that:

$$\Big| \Big\langle \frac{\partial p_\infty}{\partial n}, \phi \Big\rangle_{\Gamma_\infty} \Big| = \Big| \Big\langle \frac{\partial h}{\partial n}, p_\infty \Big\rangle_{\Gamma_3} - \Big\langle \frac{\partial p_\infty}{\partial n}, h \Big\rangle_{\Gamma_3} \Big| \leq$$

$$\leq C \|p_\infty\|_{2,D_1} \|h\|_{2,D_1} \leq C \|p_\infty\|_{0,\Omega_4} \|h\|_{0,\Omega_4} \leq C \|p_\infty\|_{0,\Omega_4} \|h\|_{1,\Omega_4} . \quad (1.5.22)$$

On the other hand, recalling (1.5.21) and further assuming that $\phi = 0$ in $\Omega_3 \setminus \overline{\Omega}_2$, we have that:

$$\|h\|_{1,\Omega_4} \leq \|\phi\|_{1,\Omega_4} + \|\psi\|_{1,\Omega_4} \leq \|\phi\|_{1,\Omega_\infty \setminus \overline{\Omega}_2} + \|\psi\|_{1,\Omega_\infty \setminus \overline{\Omega}_2} \leq C \|\phi\|_{1,\Omega_\infty \setminus \overline{\Omega}_3} . \quad (1.5.23)$$

We can now estimate the fractional norm. Since ϕ is in particular an arbitrary function of $H^1_{\Gamma_3}(\Omega_\infty \setminus \overline{\Omega}_3)$, we have that:

$$\Big\| \frac{\partial p_\infty}{\partial n} \Big\|_{-1/2,\Gamma_\infty} = \sup_{\phi} \frac{\Big| \Big\langle \frac{\partial p_\infty}{\partial n}, \phi \Big\rangle_{\Gamma_\infty} \Big|}{\|\phi\|_{1/2,\Gamma_\infty}} \leq C \|p_\infty\|_{0,\Omega_4} .$$

Finally, since the load function Φ is arbitrary as well, we can conclude that

$$\|p_\infty\|_{0,\Omega_4} = \sup_{\Phi} \frac{|(\Phi, p_\infty)|}{\|\Phi\|_{0,\Omega_4}} \leq C \, e^{-\sigma_0 k R} \|p_\infty\|_{0,\Omega_4} ,$$

hence the factor $C\,e^{-\sigma_0 kR}$ is less than one, provided that R is sufficiently large. This implies that $p_\infty = 0$ on Ω_4. By unique continuation it follows that p_∞ vanishes on all Ω_∞. ■

We can now prove the following convergence result.

Theorem 1.5.5. *The truncated PML problem (1.5.17) is well posed, provided that R is sufficiently large, and the following estimate holds:*

$$\|p - p_\infty\|_{0,\Omega_4} \le C\,e^{-2\sigma_0 kR}\|p_0\|_{1/2,\Gamma}\ .$$

Proof. As anticipated, the uniqueness result of Theorem 1.5.4 and the Fredholm alternative imply existence of the solution p_∞ to the truncated problem. In order to prove the estimate we proceed in a similar fashion as before. Let us recall the auxiliary problem (1.5.19) from the proof of Theorem 1.5.4. Its solution p_Φ is an element of $H^1_0(\Omega^c)$ and its restriction to the set Ω_∞ belongs to the space $H^1_\Gamma(\Omega_\infty)$ and satisfies $-\tilde{\Delta} p_\Phi - k^2 p_\Phi = \Phi$ in Ω_∞. The error function $e = p - p_\infty$ is an element of $H^1_\Gamma(\Omega_\infty)$ and satisfies the homogeneous problem

$$\begin{aligned} -\tilde{\Delta} e - k^2 e = 0 \qquad & \text{in } \ \Omega_\infty\ , \\ e = p \qquad & \text{on } \Gamma_\infty\ . \end{aligned}$$

Applying integration by parts twice, we obtain that

$$(\Phi, e) = \tilde{b}_\infty(e, p_\Phi) - \Big\langle \frac{\partial p_\Phi}{\partial n}, p \Big\rangle_{\Gamma_\infty} = \Big\langle \frac{\partial e}{\partial n}, p_\Phi \Big\rangle_{\Gamma_\infty} - \Big\langle \frac{\partial p_\Phi}{\partial n}, p \Big\rangle_{\Gamma_\infty}\ . \qquad (1.5.24)$$

The first term on the right hand side is estimated as in (1.5.20), with p_∞ replaced by e:

$$\left| \left\langle \frac{\partial e}{\partial n}, p_\Phi \right\rangle_{\Gamma_\infty} \right| \leq C\, e^{-\sigma_0 kR} \|\Phi\|_{0,\Omega_4} \left\| \frac{\partial e}{\partial n} \right\|_{-1/2,\Gamma_\infty} . \tag{1.5.25}$$

In order evaluate the fractional norm $\|\partial e/\partial n\|_{-1/2,\Gamma_\infty}$, we follow the same approach of Theorem 1.5.4. For every function ϕ belonging to $H^1_{\Gamma_2}(\Omega_\infty \setminus \overline{\Omega}_2)$ we consider the auxiliary problem:

Find $\psi \in H^1_0(\Omega_\infty \setminus \overline{\Omega}_2)$ such that:

$(\nabla\psi, \nabla q) - k^2 d_0^2\, (\psi, q) = (\nabla\phi, \nabla q) - k^2 d_0^2\, (\phi, q)$ for all $q \in H^1_0(\Omega_\infty \setminus \overline{\Omega}_2)$.

As previously discussed, this problem is well posed, $\psi = \phi$ on $\Omega_\infty \setminus \overline{\Omega}_2$, and the estimate $\|\psi\|_{1,\Omega_\infty\setminus\overline{\Omega}_2} \leq C\|\phi\|_{1,\Omega_\infty\setminus\overline{\Omega}_2}$ holds. We set $h = \phi - \psi$ and use the second Green's identity as before:

$$0 = \int_{\Omega_\infty\setminus\overline{\Omega}_3} (e\,\Delta h - h\,\Delta e) = \left\langle \frac{\partial h}{\partial n}, e \right\rangle_{\Gamma_\infty} + \left\langle \frac{\partial h}{\partial n}, e \right\rangle_{\Gamma_3} - \left\langle \frac{\partial e}{\partial n}, h \right\rangle_{\Gamma_\infty} - \left\langle \frac{\partial e}{\partial n}, h \right\rangle_{\Gamma_3} .$$

Recalling that $h = \phi$ on Γ_∞ and that $e = p$ on Γ_∞, we have that:

$$\left| \left\langle \frac{\partial e}{\partial n}, \phi \right\rangle_{\Gamma_\infty} \right| = \left| \left\langle \frac{\partial h}{\partial n}, p \right\rangle_{\Gamma_\infty} \right| + \left| \left\langle \frac{\partial h}{\partial n}, e \right\rangle_{\Gamma_3} - \left\langle \frac{\partial e}{\partial n}, h \right\rangle_{\Gamma_3} \right| .$$

Let's work on the first term on the right hand side. Since h solves the Helmholtz equation in $\Omega_\infty \setminus \overline{\Omega}_2$, we can use the standard trace inequality $\|\partial h/\partial n\|_{-1/2,\Gamma_\infty} \leq C\|h\|_{1,\Omega_\infty\setminus\overline{\Omega}_2}$. Similarly, applying Lemma 1.5.2, with a subdomain D_1 enclosing Γ_∞ and such that $\overline{D}_1 \subset S_\varepsilon$, and Lemma 1.5.3 to p, we obtain that $\|p\|_{1/2,\Gamma_\infty} \leq C\, e^{-\sigma_0 kR}\|p\|_{0,\Omega_4}$. Finally, recalling that $\|h\|_{1,\Omega_\infty\setminus\overline{\Omega}_2} \leq C\|\phi\|_{1,\Omega_\infty\setminus\overline{\Omega}_2}$, we obtain that:

$$\left| \left\langle \frac{\partial h}{\partial n}, p \right\rangle_{\Gamma_\infty} \right| \leq \left\| \frac{\partial h}{\partial n} \right\|_{-1/2,\Gamma_\infty} \|p\|_{1/2,\Gamma_\infty} \leq C\, e^{-\sigma_0 kR} \|p\|_{0,\Omega_4} \|\phi\|_{1,\Omega_\infty\setminus\overline{\Omega}_2} . \tag{1.5.26}$$

In order to bound the second term, we proceed as in (1.5.22) and (1.5.23), with p replaced by e, to obtain

$$\Big|\Big\langle\frac{\partial h}{\partial n}, e\Big\rangle_{\Gamma_3} - \Big\langle\frac{\partial e}{\partial n}, h\Big\rangle_{\Gamma_3}\Big| \leq C\|e\|_{0,\Omega_4}\|\phi\|_{1,\Omega_\infty\setminus\overline{\Omega}_2} \,.$$

We can finally evaluate the fractional norm:

$$\Big\|\frac{\partial e}{\partial n}\Big\|_{-1/2,\Gamma_\infty} = \sup_{\phi}\frac{\Big|\Big\langle\dfrac{\partial e}{\partial n}, \phi\Big\rangle_{\Gamma_\infty}\Big|}{\|\phi\|_{1/2,\Gamma_\infty}} \leq C\big(e^{-\sigma_0 kR}\|p\|_{0,\Omega_4} + \|e\|_{0,\Omega_4}\big)\,. \tag{1.5.27}$$

The second term on the right hand side of (1.5.24) is again estimated in a similar fashion as $\langle\partial h/\partial n, p\rangle_{\Gamma_\infty}$ in equation (1.5.26). In this case, since both p_Φ and p satisfy the Helmholtz equation in $\Omega^c\setminus\overline{\Omega}_2$, we obtain a double rate of convergence:

$$\begin{aligned}\Big|\Big\langle\frac{\partial p_\Phi}{\partial n}, p\Big\rangle_{\Gamma_\infty}\Big| \leq \Big\|\frac{\partial p_\Phi}{\partial n}\Big\|_{1/2,\Gamma_\infty}\|p\|_{1/2,\Gamma_\infty} &\leq C\|p_\Phi\|_{0,S_\varepsilon}\|p\|_{0,S_\varepsilon} \leq \\ &\leq C\,e^{-2\sigma_0 kR}\|p\|_{0,\Omega_4}\|p_\Phi\|_{0,\Omega_4} \leq C\,e^{-2\sigma_0 kR}\|p\|_{0,\Omega_4}\|\Phi\|_{0,\Omega_4}\,, \end{aligned}\tag{1.5.28}$$

where, as before, D_1 encloses Γ_∞ and $\overline{D}_1 \subset S_\varepsilon$. Combining together equation (1.5.24), (1.5.25), (1.5.27) and (1.5.28) we obtain

$$\begin{aligned}|(\Phi, e)| &\leq \Big|\Big\langle\frac{\partial e}{\partial n}, p_\Phi\Big\rangle_{\Gamma_\infty}\Big| + \Big|\Big\langle\frac{\partial p_\Phi}{\partial n}, p\Big\rangle_{\Gamma_\infty}\Big| \\ &\leq C\,e^{-\sigma_0 kR}\|\Phi\|_{0,\Omega_4}\Big\|\frac{\partial e}{\partial n}\Big\|_{-1/2,\Gamma_\infty} + C\,e^{-2\sigma_0 kR}\|p\|_{0,\Omega_4}\|\Phi\|_{0,\Omega_4} \\ &\leq C\,e^{-\sigma_0 kR}\|\Phi\|_{0,\Omega_4}\big(e^{-\sigma_0 kR}\|p\|_{0,\Omega_4} + \|e\|_{0,\Omega_4}\big)\,.\end{aligned}$$

This implies that

$$\|e\|_{0,\Omega_4} \leq C\big(e^{-\sigma_0 kR}\|e\|_{0,\Omega_4} + e^{-2\sigma_0 kR}\|p\|_{0,\Omega_4}\big)\,,$$

hence, provided that R is sufficiently large, we have that:

$$\|p - p_\infty\|_{0,\Omega_4} \leq Ce^{-2\sigma_0 kR}\|p\|_{0,\Omega_4} \, .$$

Finally, recalling a standard trace inequality we have that $\|p\|_{1,\Omega^c} \leq C\|p_0\|_{1/2,\Gamma}$. This implies the desired estimate. ■

Chapter 2

Finite Element Code Design

2.1 Geometry Modeling

Geometry modeling is the very much neglected foundation of any finite element code. In our finite element library, it is handled by the *Geometry Modeling Package* (GMP). Our technology builds on the very well known concept of the *Mesh Based Geometry* (MBG) description, discussed extensively in, *e.g.*, [21, 25]. The object of interest is partitioned into a mesh-like structure consisting of blocks of the same shapes as those used for finite elements, namely prisms, hexahedra, tetrahedra and pyramids for three dimensional problems, and quadrilaterals and triangles for two dimensional applications. Each of the blocks comes with a parametrization, i.e., a function $\mathbf{x}_b(\eta_1, \eta_2, \eta_3)$ mapping a corresponding reference block onto the particular block within the physical domain. These parametrizations must be at least C^0-compatible. Intuitively speaking, if the reference prisms, hexahedra, tetrahedra and pyramids are covered with uniform meshes and identical number of partitions—in fact elements!—along the edges, the geometry maps will map those uniform meshes onto a regular finite element mesh in the physical domain. This is precisely the concept behind the classical algebraic mesh generators.

The ultimate goal of GMP is to return values of the parameterizations $\mathbf{x}_b$ and their derivatives. There are two fundamental technical difficulties in constructing a MBG description for an object. The first one deals with partitioning of the object into blocks and setting appropriate connectivities – the task encountered in any unstructured mesh generation. While for academic geometries consisting just of a few blocks, this can be done "by hand", for most practical problems, we have to resort to third party mesh generators like Netgen [70] that we have used in this project. The second technical task deals with the construction of compatible parametrizations. The main idea here is to employ a *bottom-up* approach used for example in unstructured mesh generation. Besides the geometrical blocks, we introduce geometrical entities of lower dimension: points (vertices), (segments of) curves, and quads and triangles. Having defined the points, we define first (parametrizations for) the curves connectining the points. Next, we construct parametrizations for quadrilateral and triangular faces in such a way that they match the already defined parameterizations for edges. Mathematically speaking, given a quad or triangle, the task is to extend parametrization of its edges to the interior of the figure. This is done using techniques such as transfinite interpolation and implicit parameterizations. For C^0-conforming parametrizations, the task is relatively easy; if we demand more global regularity, these constructions become quite technical, see *e.g.*, [23]. Finally, having constructed the parametrizations for faces, we use again the transfinite parameterization techniques to extend them to the whole blocks.

To describe the domain of interest for this project, we have interfaced with Joachim Schöeberl's tetrahedral mesh generator Netgen [70]. Netgen takes a few input formats; for objects with easy to moderate geometrical complexity it is highly convenient to use the *Constructive Solid Geometry* (CSG) description of the geometrical object. The head model we have been studying is indeed a CSG object. Given a number of algebraic primitives like half-space, interior or exterior of a sphere, cylinder or cone, etc., we use Boolean operations to define the object. Netgen then identifies the resulting 2D polygonal interfaces, 1D edges and vertices, and uses *Delaunay* and *Advancing Front* mesh generation techniques to generate a mesh consisting of linear tetrahedra. The mesh generation is done using the bottom-up approach. First, nodes are placed at CSG vertices, then 1D meshes are generated along the CSG edges, followed by generation of triangular meshes on interfaces and, finally, generation of tetrahedral meshes inside of the resulting 3D polyhedral domains. Netgen generates meshes with a minimum number of tetrahedra implied by geometrical scales. This fits perfectly the philosophy of MBG description. When dealing with “real” objects, data coming from an MRI scan are usually the available descriptions; they easily translate into a faceted representation of the bounding surfaces of the object. We emphasize that obtaining a model described purely in terms of faceted surfaces implies a number of non-trivial manipulations of the geometry that goes beyond the scope of this work. Nevertheless, if such a model were available to us, it would be possible to use it in our computations.

The CSG model already involves a number of manipulations of the geometry. An attempt to mesh thin-walled structures like membranes present in this project, with tetrahedral meshes leads to a prohibitive number of elements and distorted tets. To avoid the problem, the membranes are first identified as 2D interfaces, forcing Netgen to generate meshes conforming to the membranes and only then extruded into thin 3D objects. This implies that our FE code must be able to manipulate, at least to some extent, the geometry returned by the external mesh generator, in order to make it suitable for computations. More precisely, defining a membrane involves setting up the surface occupied by the membrane, a set of surfaces bounding the actual membrane, and an additional set of surfaces bounding the extrusion domain. The simplest example is provided by extruding a membrane inside of a cylindrical shell, see Figure 2.1. The membrane is first identified with a plane cutting through the cylinder. The inner cylinder bounds the actual extruded 3D membrane, while the outer cylinder terminates the set of extruded triangles. All vertex points in the interior of the 2D membrane are extruded into line segments, whereas the points on the outer cylinder stay unduplicated. All triangles within the inner cylinder are extruded into prisms. Triangles within the cylinder wall may be extruded into prisms, tets or pyramids, depending upon the number of triangle vertices located on the outer cylindrical surface.

In this way, the extrusion process has forced us to introduce into the code both prisms and pyramids. The prisms are also the element of choice for discretizing problems with spherical geometry (used for code verification in this project) and implementation of *Perfectly Matched Layer.* In both cases, we start with a triangular mesh on a sphere and extrude it in the radial direction into a number of layers of prismatic elements.

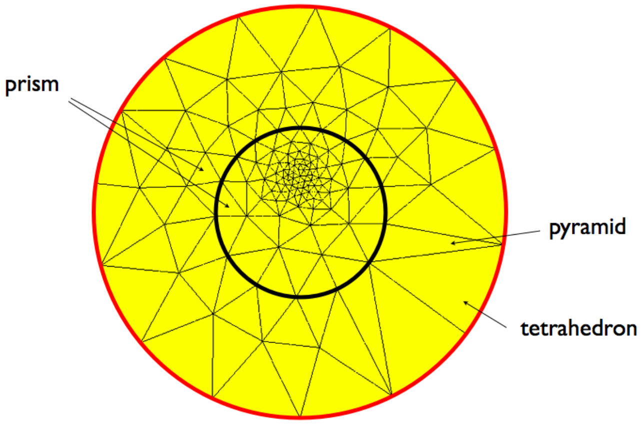


Figure 2.1: Extrusion of a membrane inside a cylindrical shell. The red circle represents the bounding cylinder, while the black circle represents the conforming cylinder.

2.2 Shape Functions

The construction of shape functions is a fairly trivial task for elements with a tensor product structure, such as the quad and the hexahedron: given a space of 1D shape functions, *e.g.*, Legendre polynomial or integrated Legendre polynomials, the shape functions are simply tensor products of 1D shape functions. This construction has been described in many works, among which we recall the famous book *Finite Element Analysis* by Szabó and Babuška, see [80], and a paper by Szabó, Düster and Rank, [79]. A fairly well established way of constructing triangle shape functions is by means of the Duffy's transformation that maps the master square onto the master triangle by collapsing one edge; this idea was originally proposed by Dubiner in [30]. Successively, Karniadakis and Sherwin in [49] have employed the same idea to extended the construction to tetrahedra; more recently, in [86], Zaglmayr used the same approach to construct shape functions for the prism. Shape functions that are constructed via the Duffy's transformation are automatically compatible between elements of different shapes and enjoy fast integration properties; more precisely, for elements with a tensor product structure, an $\mathcal{O}(p^7)$ fast integration algorithm can be implemented, as discussed in [21]. In [80], Szabó and Babuška describe a construction of shape functions on the master triangle that does not rest on the Duffy's transformation; Devloo, in [33], taking inspiration from the construction on master triangle, extends the construction to elements of all types, including the pyramid. A similar construction of shape functions has been derived by Solín in [77], however he does not discuss the pyramidal element. A parallel between the construction of shape functions and transfinite interpolation is found in [80] for the triangular and the quadrilateral element, and in [79] for the quadrilateral element only. Comparison between the construction of shape functions and transfinite interpolation for the 3D elements is not discussed in any of the previously cited works. Finally, the pyramid is a much neglected element; construction of shape functions have been proposed by Zaglmayr in [87], and Nigam and Phillips in [60].

In this work we are faced with the challenge of constructing hierarchical spaces of shape functions for elements of all types, including neglected pyramid, see Figure 2.2. In order to obtain an H^1-conforming FE space, such spaces must be constructed in a way that the global basis functions are continuous across interlement faces and edges. In 2D and for hexas only meshes, compatibility of element shape functions on common faces and edges is enforced by the use of sign factors originally proposed by Szabo, see [80, 21, 25]. The use of sign factors is accounted for in the assembling procedure and complicates significantly implementation of constrained approximation (hanging nodes).

The situation is much more difficult for elements with triangular faces. The inherent conflict between hierarchical construction of shape functions and irrotational invariance may result in a situation when a single face shape function for an element must be connected with all shape functions for the neighboring element, even for regular meshes. One way to avoid the problem is to set up element coordinates in a proper way. For tets only meshes, Ainsworth and Coyle [1] have shown that one can solve the problem by considering tetrahedra of two types. Implementing the procedure requires resetting tets to vertex connectivities.

In the presented implementation we have followed a more radical way proposed originally by Ph. Devloo. The edge and face basis functions are defined first on edges and faces using the edge and face parametrizations, and then extended into the neighboring elements. In other words, we first define the global basis functions and only then identify element shape functions as *restrictions* of the basis functions to the elements. This implies that the orientations (maps from local to global coordinates for edges and faces) are *accounted for in the shape functions routines.* One may think thus about not a single element but a family of elements for all possible variations of orientations. The input to element shape functions includes thus element type (prism, brick, tet, pyramid), order and orientation for element nodes and master element coordinates ξ_i of a point within the element. The routine returns

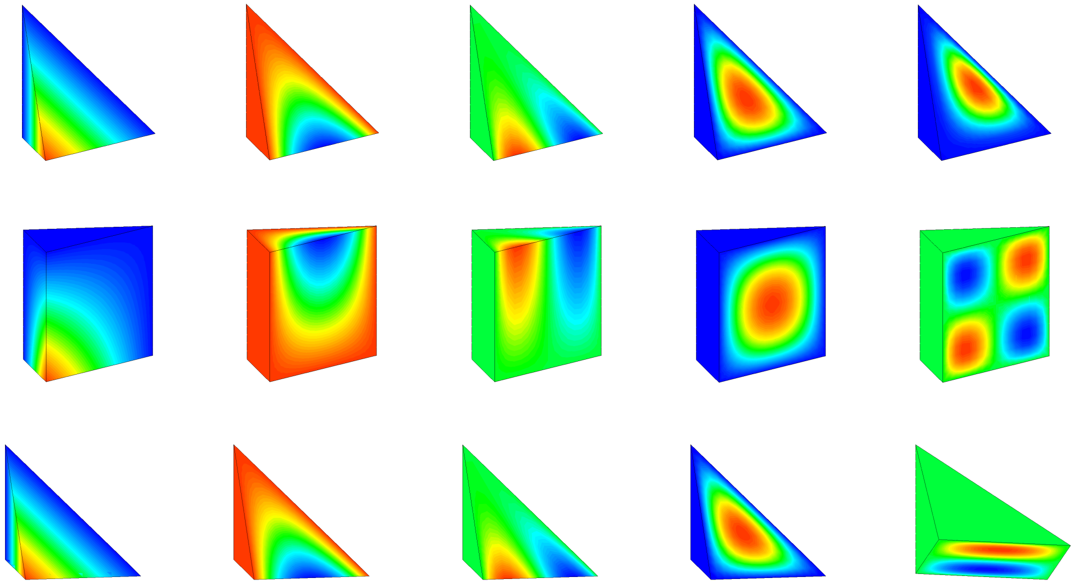


Figure 2.2: Hierarchical shape functions for elements of all shapes. Column-wise: vertex mode, even edge bubble, odd edge bubble, lowest order face bubble, higher order face bubble.

then the number of the element shape functions, their values and derivatives.

For a detailed description of the element shape functions, see [40]. For alternative constructions see the work of Solín [77], Zaglmair and Schöeberl [71] and, for pyramids, Nigam and Phillips [60].

2.3 Exact Geometry Elements & Isoparametric Elements

The code supports the use of both isoparametric and exact geometry elements. In the latter case, the GMP maps $\mathbf{x}_b$ are directly implied to compute the element stiffness matrix. In the first case, the exact geometry map is replaced by a polynomial map $\mathbf{x}_{hp}$, which is conveniently computed as a linear combination of shape functions. The coefficients of such a combination are called geometry dof's and determined by Projection Based Interpolation (PBI) of the GMP maps. We anticipate that PBI is also employed to determine Dirichlet dof's. While a detailed discussion on PBI can be found in [21, 25], we just recall its main properties.

For every energy space, a PBI is defined in a way such that it enjoys three fundamental properties: locality, global conformity and optimality. Locality refers to the fact that it is an element-wise construction, global conformity means that the interpolant of a *sufficiently* regular function is conforming in either H^1, $H(\text{curl})$ or $H(\text{div})$, according to the selected energy space. In the case of the geometry map $\mathbf{x}_{hp}$, conformity is an immediate consequence of the C^0-compatibility of the GMP parameterizations. Finally, optimality refers to the fact that the interpolation estimates are optimal with respect to both p and h. Indeed, the following p-estimates have been proved in [27]:

$$\begin{aligned}
&\|u - \Pi^{\text{grad}} u\|_{1,\Omega} &&\le C\,\frac{\ln^2 p}{p^{r-1}}\,\|u\|_{r,\Omega} &&\forall\, u \in H^r(\Omega) &&, \quad r > \frac{3}{2}\,;\\
&\|\boldsymbol{E} - \Pi^{\text{curl}} \boldsymbol{E}\|_{\text{curl},\Omega} &&\le C\,\frac{\ln p}{p^{r}}\,\|\boldsymbol{E}\|_{r,\text{curl},\Omega} &&\forall\, \boldsymbol{E} \in H^r(\text{curl},\Omega) &&, \quad r > \frac{1}{2}\,;\\
&\|\boldsymbol{v} - \Pi^{\text{div}} \boldsymbol{v}\|_{\text{div},\Omega} &&\le C\,\frac{\ln p}{p^{r}}\,\|\boldsymbol{v}\|_{r,\text{div},\Omega} &&\forall\, \boldsymbol{v} \in H^r(\text{div},\Omega) &&, \quad r > 0\,;
\end{aligned}$$

where $H^r(\text{curl},\Omega) = \{\boldsymbol{E} \in H^r(\Omega) \;:\; \text{curl}\,\boldsymbol{E} \in H^r(\Omega)\}$ endowed with the norm $\|\boldsymbol{E}\|^2_{r,\text{curl},\Omega} = \|\boldsymbol{E}\|^2_{r,\Omega} + \|\,\text{curl}\,\boldsymbol{E}\|^2_{r,\Omega}$, and $H^r(\text{div},\Omega) = \{\boldsymbol{v} \in H^r(\Omega) \,:\, \text{div}\,\boldsymbol{v} \in H^r(\Omega)\}$, endowed with the norm $\|\boldsymbol{v}\|^2_{r,\text{div},\Omega} = \|\boldsymbol{v}\|^2_{r,\Omega} + \|\,\text{div}\,\boldsymbol{v}\|^2_{r,\Omega}$.

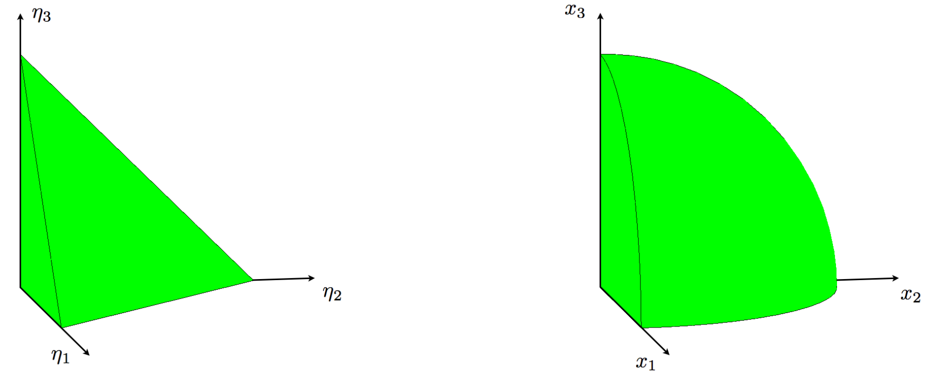
The corresponding h-estimates can be derived using standard scaling arguments. A critical point is that the PBI operators corresponding to master and physical element commute with each others. More precisely, if we let $\hat{u} = u \circ \mathbf{x}_b$, which is the so-called Piola transform for H^1, then we should require the commutativity property

$$\widehat{\Pi u} = \hat{\Pi}\hat{u} \, ,$$

where $\hat{\Pi}$ is the PBI operator on the master element. Unfortunately, the commutativity property does not hold, unless the geometry map reduces to a scaling, namely $\mathbf{x}_b = h\,\boldsymbol{\eta}$, possibly superimposed with a rigid body motion.

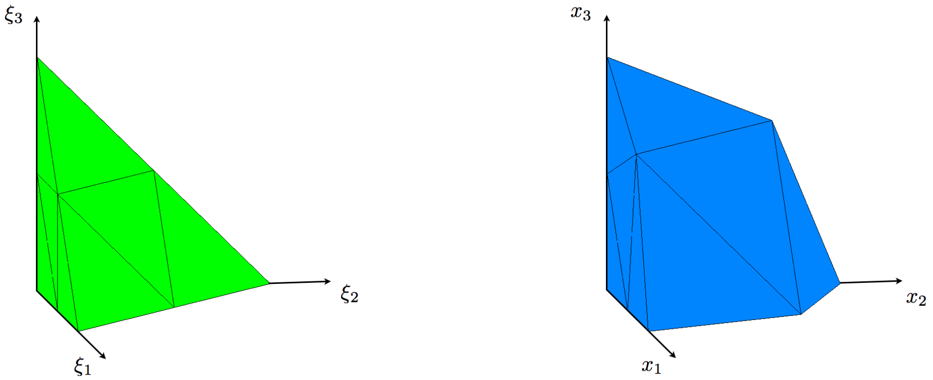
In order to make the discussion precise, we need to introduce the *reference* space. We are already familiar with the master space (η_1, η_2, η_3) – where the master tetrahedron, prism, hexahedron and pyramid live, see Figure 2.3(a) – and with the physical space (x_1, x_2, x_3) – where the physical blocks live, see Figure 2.3(b). Let's recall once again that the physical elements are images of the master element of corresponding shape under the GMP map, i.e., $\boldsymbol{x} = \mathbf{x}_b(\eta_1, \eta_2, \eta_3)$. The element refinements are performed in a third space, the reference space (ξ_1, ξ_2, ξ_3), see Figure 2.3(c). In practice, for every refinement kind, we define a refinement map from the master element to the reference space. The geometry dof's are generated by PBI in the reference space. This is essential since, in general, the refinement map is *not* a rescaling of the master element possibly composed with a rigid body motion. This is evident in the case of the isotropic refinement of the tetrahedron, see Section 2.6 for further details, shown in Figure 2.3(c). Finally, Figure 2.3(d) shows the first order *hp*-blocks that correspond to the isotropic refinement of the tetrahedron.

The FE code supports *fully* variable order elements, namely independent orders of approximations are assigned to each element's edges, faces and interior. More-



(a) master tetrahedron

(b) a tetrahedral element in physical space



(c) tetrahedral refinement in reference space

(d) *hp*-blocks resulting from refinement

Figure 2.3: Master, reference, and physical space.

over, in the case of a face or interior with a tensor product structure, independent orders of approximation are assigned to each direction. For example, a quadrilateral face has an horizontal and vertical order of approximation, similarly to prism, while a hexahedron has a horizontal, lateral and vertical order of approximation. In fact, we assign a vector-valued order of approximation

$$\mathbf{p} = \left(p_{\,\mathrm{edge}_{1,\ldots,\#\mathrm{edges}}}\ ;\ p_{\,\mathrm{face}_{1,\ldots,\#\mathrm{faces}}}\ ;\ p_{\,\mathrm{interior}}\right)$$

to each element. Let's remark that a *hierarchical* basis of shape functions is essential to support variable order finite elements.

When the geometry map $\mathbf{x}_{hp}$ and the finite element it pertains to share the same order of approximation $\mathbf{p}$—indeed the same order for edges, faces, and interior!—, we refer to such an element as *isoparametric*. The importance of isoparametric element is that they are linear complete: they reproduce linear body motions in the physical space exactly. This fact is deemed of crucial importance for the case of linear elasticity. Finally, the exact geometry map $\mathbf{x}_b$ is in general much more expensive to evaluated than the polynomial map $\mathbf{x}_{hp}$.

While each element in the mesh has *a priori* arbitrary orders of approximation for its edges, faces and interior, the ultimate goal is to build piecewise polynomial spaces W_h, Q_h, V_h, and Y_h, defined on the mesh Ω_h, such that the variable order sequence

$$\begin{tikzcd}
W_h \arrow[r, "\nabla"] \arrow[d, phantom, "\cap"] & Q_h \arrow[r, "\mathrm{curl}"] \arrow[d, phantom, "\cap"] & V_h \arrow[r, "\mathrm{div}"] \arrow[d, phantom, "\cap"] & Y_h \arrow[d, phantom, "\cap"] \\
H^1 & H(\mathrm{curl}) & H(\mathrm{div}) & L^2
\end{tikzcd}$$

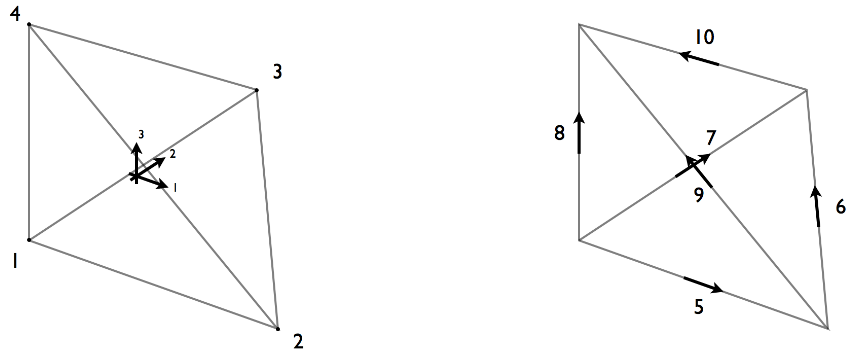
(a) Triangular element (b) Quadrilateral element

Figure 2.4: Node numbering and local orientations for 2D elements.

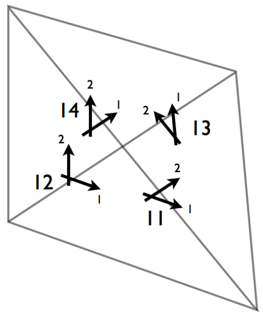
is exact. The exactness of the sequence is guaranteed by the *minimum rule*. The order of each face is set to the minimum order of the adjacent elements and, similarly, the order of each edge is set to the minimum order of all connected faces. Refer to [25] for the construction of $W_{\mathbf{p}}$, $Q_{\mathbf{p}}$, $V_{\mathbf{p}}$, $Y_{\mathbf{p}}$ for each element shape, and a detailed discussion about the exactness of the variable order sequence.

2.4 Code Data Structure

The data structure of the code contains two types of objects only: initial mesh elements, and nodes. Initial mesh elements coincide with GMP blocks. In simple words, a node is either a vertex, an (open) edge, a face (interior), or an element (interior) of the mesh. Henceforth, those node types will be referred to as vertex node, edge node, face node, and middle node. As we shall see, the middle nodes play a special role, since each one of them can be uniquely identified to an element. In the case of the (regular) initial mesh, edge nodes, face nodes and middle

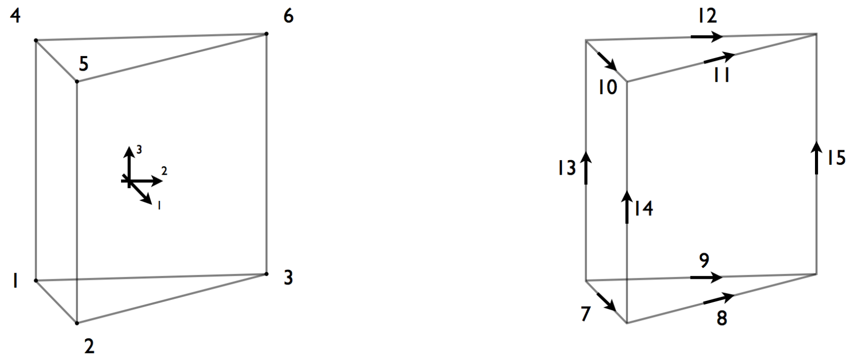


(a) Middle and vertex nodes (b) Edge nodes



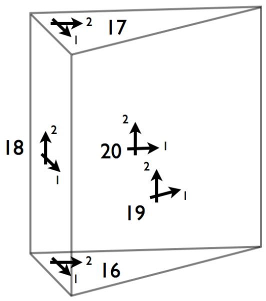
(c) Face nodes

Figure 2.5: Node numbering and local orientations for tetrahedral element.



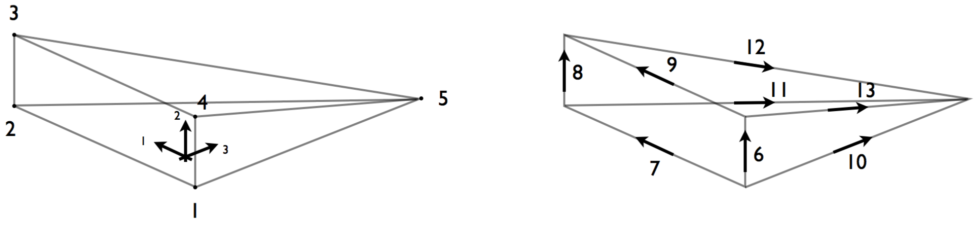
(a) Middle and vertex nodes

(b) Edge nodes



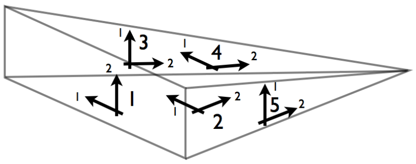
(c) Face nodes

Figure 2.6: Node numbering and local orientations for prismatic element.



(a) Middle and vertex nodes

(b) Edge nodes local orientations



(c) Face nodes local orientations

Figure 2.7: Node numbering and local orientations for pyramidal element.

nodes inherit *global* orientations from the corresponding geometrical entities, i.e., curves, triangles or quads, and GMP blocks respectively. The orientation of the element, which naturally coincides with the one of the middle node, induces a local enumeration of the element nodes (vertices, edges, faces, and middle) as well as a *local* orientation on its edges and faces. A similar reasoning holds for triangular and quadrilateral faces; in this case the global orientation coincides with the one of the face node and induces a local enumeration of the face nodes (vertices and edges) and a local orientation on the edges. Local nodes numbering and orientations for all elements are illustrated in Figure 2.4, 2.5, 2.6, and 2.7. For a more detailed discussion about orientations see [40].

Informations about nodal connectivities and local *to* global orientations are explicitly stored in the initial element data type. While a minus sign in front of an edge node number indicates opposite orientations, a digit is appended to a face node number to account for rotations and flips between the local and global system of coordinates. In fact, what is recorded in the data structure is a face label generated as:

$$\texttt{face label} = \texttt{node number*10} + \texttt{orientation flag}\,.$$

Orientations flags for triangular and quadrilateral faces are illustrated in Figure 2.8 and Figure 2.9 respectively.

As we will see in details in Section 2.5, the nodal connectivities for an element (not necessarily an initial mesh element) are reconstructed on-the-fly rather then being explicitly stored in the data structure. Beside the advantage of memory saving, this approach implies that in the process of refining an element we do not

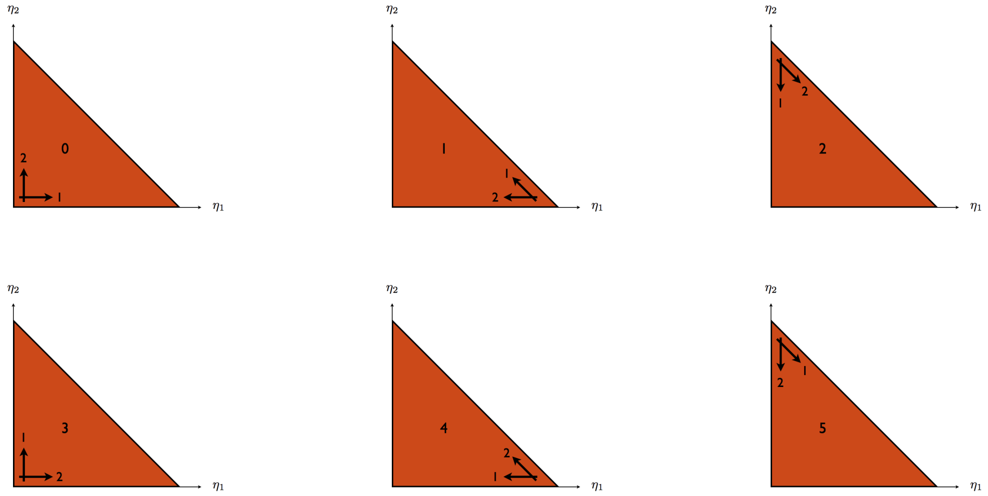


Figure 2.8: Triangle orientations.

have to worry about determining nodal connectivities and orientations for the new elements that were generated in the refinement process. In fact, given an element, its refinement history along with the information stored in the initial mesh element data type are sufficient to "put back together" all of its pieces, i.e., the nodes, in the appropriate fashion, i.e., according to their orientations.

We call the process of filling up the FE code data structure "initial mesh generation." This task is performed using both informations coming from the GMP and informations about multi-physics, initial orders of approximations, boundary and interface conditions. As anticipated, every initial mesh element can be identified to its unique middle node; we take advantage of this fact by generating middle nodes first, so that their numbering coincides with the one of initial elements. Thus, interfacing between initial mesh elements and initial mesh middle nodes is completely

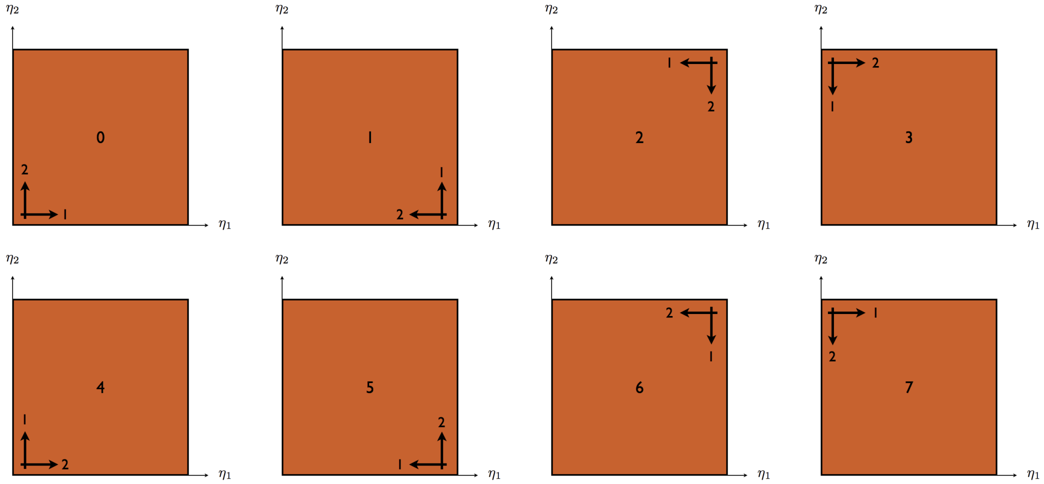


Figure 2.9: Quadrilateral orientations.

trivial. After mesh generation, we systematically identify an element to its interior node. The context shall imply whether a middle node is understood as a whole element, or simply a node *per se*.

Writing a multi-physics code implies a number of design challenges. Quantities of interest, that we call physical attributes, will *a priori* belong to different energy spaces, be scalar-valued or vector-valued, require different boundary and interface conditions to be imposed on each one of their components. For example, in the case of an acoustics/elasticity problem, we have a scalar-valued H^1 physical attribute, the acoustic pressure, and a vector-valued H^1 physical attribute, the elastic displacement. On the other hand, if we were to solve an electromagnetic problem, we would have one scalar-valued $H(\text{curl})$ attribute, namely the electric field.

Let's consider again the acoustics/elasticity problem. While an element belongs to either the acoustic or the elastic subdomain, it should *a priori* support both attributes. Indeed, in the case of an acoustic element that is adjacent to the elastic subdomain, the element stiffness matrix incorporates off-diagonal blocks that represent the interaction between acoustics and elasticity. Thus, all quantities of interest, must be supported simultaneously throughout the whole domain. Each node either supports a physical attributes or does not, which leads to $2^{\#\,\text{attributes}}$ possible scenarios. By indicating the presence or absence of a physical attribute with 1 or 0 respectively, we can encode this information into a single digit, that we call node *case*, using a binary system.

The initial mesh element data type includes the following attributes:

- type: hexahedron, tetrahedron, prism, pyramid
- list of physical attributes supported by the element middle node
- boundary and interface conditions flags for each supported physical attribute
- nodal connectivities (vertices, edges, faces, middle), including edges and faces orientations
- neighbors across faces
- corresponding GMP block

In general terms, the element data type stores full information about the initial mesh nodal connectivity and the supported physical attributes. The block number, which is in fact a nickname since elements of different shapes are allowed, constitutes the

`0`	no boundary condition
`1`	Dirichlet boundary condition on all components of attribute
`6`	Dirichlet boundary condition on 1^{st} component of an H^1 attribute
`7`	Dirichlet boundary condition on 2^{nd} component of an H^1 attribute
`8`	Dirichlet boundary condition on 3^{rd} component of an H^1 attribute

Table 2.1: Explanation of boundary condition flags.

interface between the GMP and the FE library. The boundary and interface condition flags (for each attribute supported by the *element*, one flag per face is needed) are problem dependent and need to be set up by the user. A list of reserved values is found in Table 2.1.

In order to present the node data type, we need to anticipate that an element is refined by generating son nodes for all of its nodes, see Section 2.6 for a detailed discussion. This naturally leads to introduce father-to-son relations between the nodes and to organize the nodes database in a tree. The node data type includes the following attributes:

- type: vertex, edge, middle of a triangle or quadrilateral, middle of a hexahedron, tetrahedron, prism or pyramid
- case
- order of approximation
- boundary and interface conditions flag

- refinement flag
- pointers to father and sons
- activation flag indicating whether the node is constrained or not
- solution degrees of freedom
- geometry degrees of freedom

The node's boundary and interface condition flag is derived from the element's flag by enforcing the so called *hierarchy* of Dirichlet boundary conditions. In simple terms, when a Dirichlet boundary condition is imposed on a element's face, it will imply a Dirichlet boundary condition on the face node, as well as adjacent edge nodes and vertex nodes.

The refinement flag indicates the node refinement kind. Let's discuss the activation flag. A *constrained node* is a node whose solution dof's are obtained as a linear combination of the solution dof's of a set of nodes from the previous generation, namely a superset of the father node. We keep track of this through the activation flag: unconstrained nodes are marked as active, while constrained nodes are marked as inactive. During mesh generation, namely the initial generation of the data structure employing informations coming from the GMP, all nodes are marked as active. This is consistent with the fact that the initial mesh is regular, and constrained nodes are a generalization of the concept of hanging nodes. During mesh generation middle nodes are the first ones to be created. This allows us to trivially handle node-to-element connectivities, since the number of an initial mesh middle node coincides with the number of the corresponding initial mesh element. Finally, the geometry degrees of freedom allow for support of isoparametric elements.

2.5 Element to Nodes Connectivities

As discussed above, an element can be understood as a collection of nodes: vertices, edges, faces and middle. An element is broken by generating son-nodes for *all* of its nodes, beside vertices. In practice, new nodes are added to the database and organized into a father-to-son tree structure. In order to implement such a strategy, it is essential to be able to determine nodal connectivities for each element or, equivalently, middle node. Rather then explicitly storing connectivities in the data structure, we reconstruct them on-the-fly employing connectivities of initial mesh elements and the refinement history.

Reconstruction of nodal connectivities is the foundation of the refinement package. In order to discuss the challenges that we are faced with, let's consider the simple case of a hexahedron that we want to refine horizontally. The main difficulty of performing the wanted refinement arises from the fact that the element has completely independent global orientations associated to its interior, faces and edges. Hence, the wanted horizontal refinement corresponds to refinement of the vertical edges, and either horizontal or vertical refinements of the lateral faces, depending upon their orientations. Our over all logic is such that, given a wanted refinement of an element according to its (middle node) orientation, we partition such element into its interior, faces and edges, and break each geometrical entity *independently.* The non-trivial part is to determine the appropriate refinement kinds for the faces—this is trivial for the edges, since only one refinement kind exists—, and, so to speak, to be able to put all of the pieces back together.

We keep track of nodal connectivities and orientations using the refinement history and employing masks that provide all of the necessary informations about each type of element refinement. In practice, for each three-dimensional element type and refinement kind, we rely on three arrays:

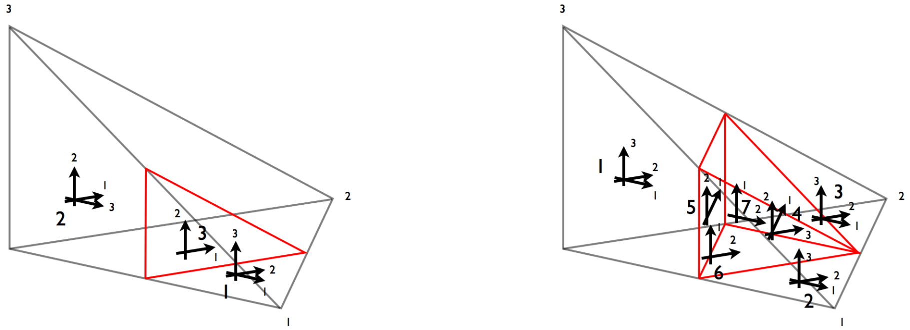
`FATHER(:,:)` , `SON(:,:)` , `ORIENTATION(:,:)` .

Those arrays have been generated by an external program through search algorithms. In order to precisely describe the meaning of each tensor, we briefly discuss node breaking. We say that a node has been broken when it is connected to son-nodes. The key point is that the orientation of the father-node induces a local enumeration of the son-nodes, as well as an orientation for each son-node. For example, when an edge node is broken, it generates two son edge nodes and one son vertex node. It seems natural to choose consistent orientations between father and son edge nodes. When it comes to face node breaking and middle node breaking things work exactly in the same way, however choices of orientations become pretty much arbitrary. Some node breaking schemes are illustrated in Figure 2.10. Notice that interior nodes are always listed first in the enumeration of son-nodes.

All of the entries `FATHER(i,j)`, `SON(i,j)`, `ORIENTATION(i,j)` refer to an attribute of the `i`-th node of the `j`-th son-element. The nodes are listed in the usual order: vertices, edges, faces and interior. By `j`-th son-element we refer to the element associated to the `j`-th son middle node. Since in the node breaking schemes middle

(a) Triangle refinement `1`

(b) Quad refinement `11`



(c) Tetrahedron refinement `32`

(d) Tetrahedron refinement `24`

(e) Prism refinement `01`

(f) Prism refinement `10`

Figure 2.10: Some node refinement kinds, showing the enumeration and orientations of son nodes. The small numbers next to the vertices indicate the orientation of the father node.

nodes are always listed first, the son-element numbers indeed start from 1. The three arrays contain the following information:

`FATHER` father-node number, according to father-*element* enumeration of its nodes.

`SON` node number, according to father-*node* enumeration of its son-nodes. Entry is 0 if the node is shared with father-element.

`ORIENTATION` node orientation, with respect to the orientation induced on it by the breaking of its father-node.

We provide two examples in order to better explain the meaning of those arrays. Let's start by considering the isotropic refinement of a triangle, see Figure 2.10(a), and focus on the first son-element, which is the small triangle in the bottom left corner. The father-element, namely the big triangle, is a collection of nodes (vertices, edges and middle); a few of those nodes are indicated in green in Figure 2.11. The local enumeration of the nodes of the son-element is indicated by the blue numbers, while the big black number refer to local enumeration of son-nodes as dictated by the father node. For example the horizontal edge of the small triangle is the first son of the horizontal edge of the big triangle and so on. The picture implies that:

`FATHER(1,1)=1` , `FATHER(2,1)=4` , `FATHER(3,1)=6` , `FATHER(4,1)=4` .

Similarly, we have that:

`SON(1,1)=0` , `SON(2,1)=3` , `SON(3,1)=3` , `SON(4,1)=1` .

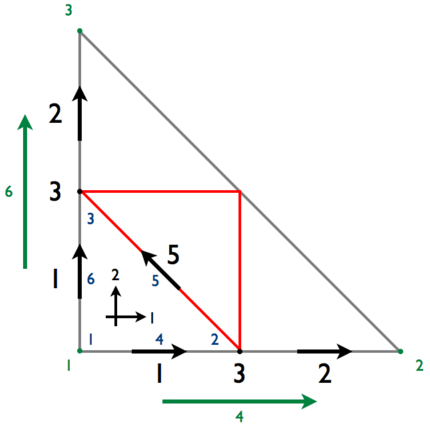


Figure 2.11: Isotropic refinement of a triangular face. The local enumeration of the nodes of the first son element is indicated in blue. Some of the nodes of the father element are indicated in green. The black numbers are the local enumeration of the son-nodes, as dictated by the father node.

It is important to remark that the 3 arrays are only associated to 3D elements only, namely a `FATHER` array associated to the isotropic refinement of the master triangle does not exists. We resorted to a 2D example purely for ease of explanation.

In order explain the meaning of the `ORIENTATION` tensor we need to resort to a more elaborate example. Let us consider a prism that has been refined vertically. The face node corresponding to the bottom triangular face is broken into 7 son-nodes, as in Figure 2.10(a): 4 triangular face nodes and 3 edge nodes. The interior node is broken into 7 son-nodes as in Figure 2.10(f): 4 prismatic interior nodes and 3 rectangular face nodes. Let us focus on the son prism in the middle, namely the 4-th son node of the interior node, hence `j=4`, which is outlined in red in Figure 2.10(f).

Its orientation, which is dictated by the refinement kind of the father node, induces *local* orientations onto its edges and faces. The local orientations of the bottom face and its edges, and the vertical faces are indicated in blue in Figure 2.12(a). The local orientations are not necessarily consistent with those induced by the breaking of the father node. As seen in figure 2.10(f), the vertical faces are son nodes of the interior node. According to the son prism enumeration of nodes, they are nodes 18, 19, 20, see Figure 2.6(c). As it can be seen from Figure 2.12(a), we have that

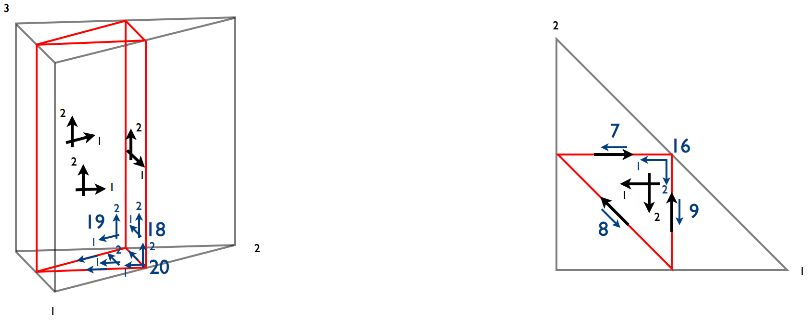
`ORIENTATION(18,4)=5` , `ORIENTATION(19,4)=5` , `ORIENTATION(20,4)=5` .

A similar reasoning holds for the bottom triangular face of the son prism. Such face and its edges are son nodes of the triangular face node corresponding to the bottom face of the father prism. The discrepancy between inherited orientations and local orientation can be seen in Figure 2.12(b). The orientation of the face node is consistent with the one induced by the breaking of the father node, `ORIENTATION(16,4)=0`, while the edge nodes have opposite orientations. Inside `ORIENTATION`, opposite edge orientations are indicated by `1`, while coinciding orientations are indicated by `0`, hence we have that:

`ORIENTATION(7,4)=1` , `ORIENTATION(8,4)=1` , `ORIENTATION(9,4)=1` .

While setting up orientations for the son-nodes is completely arbitrary, it is natural to choose orientations in a way such that the number of non-zero entry in `ORIENTATION` is minimized.

Let's see how those three arrays are employed to determine the full nodal



(a) Discrepancy for lateral faces

(b) Discrepancy for bottom face and its edges

Figure 2.12: Discrepancy between son-element local orientations (indicated in blue) and orientations as inherited by the father node (indicated in black).

connectivities `node(:)` and orientations `orientation(:)` of an element associated to a given middle node.

Step 1. Go up the tree. Record father-node and determine local son-node number, i.e., its position on the list of the son-nodes. This is the `j` index. Set `son` ← `father` and iterate the process until an initial mesh interior node is reached.

Step 2. Fetch initial mesh element connectivities and orientations. For the initial mesh ancestor, record full nodal connectivities `node_fath(:)` (vertices, edges, faces, middle) and orientations `orientation_fath(:)`, which are stored in the initial mesh element data type.

Step 3. Go down the tree. For each generation, loop over the nodes of the `j`-th son-element. For each node, indicated with index `i`, set

`ip=FATHER(i,j)` , `is=SON(i,j)` , `or=ORIENTATION(i,j)` .

If `is=0`, then the node is shared with the father element and we simply record its number and orientation:

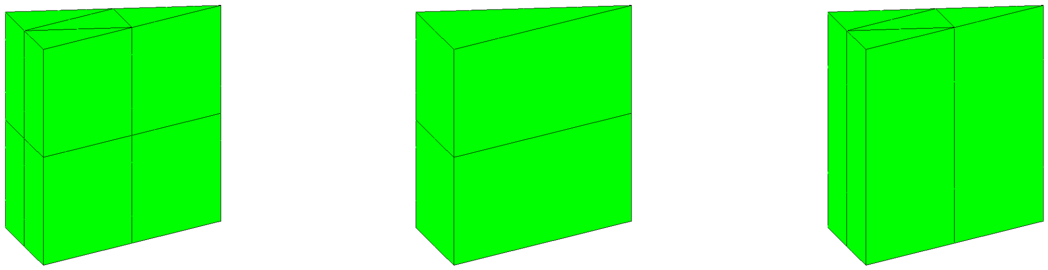
`node(i)=node_fath(ip)` , `orientation(i)=orientation_fath(ip)` .

Otherwise, we employ the element refinement kind and the relative orientation `or` of the parent node to pick the appropriate son node and determine its orientation.

2.6 Mesh Refinements

The code supports the standard refinements of a hexahedron (into two, four or eight subhexahedra), along with the prism, tetrahedron and pyramid refinements shown in Figure 2.13. The choice of anisotropic refinements for the tetrahedron and the pyramid is dictated by the fact that we do not want to refine thin-walled structures across their thickness. In practice this is achieved by introducing a refinement filter and only allowing certain refinements for elements located in a thin-walled structure.

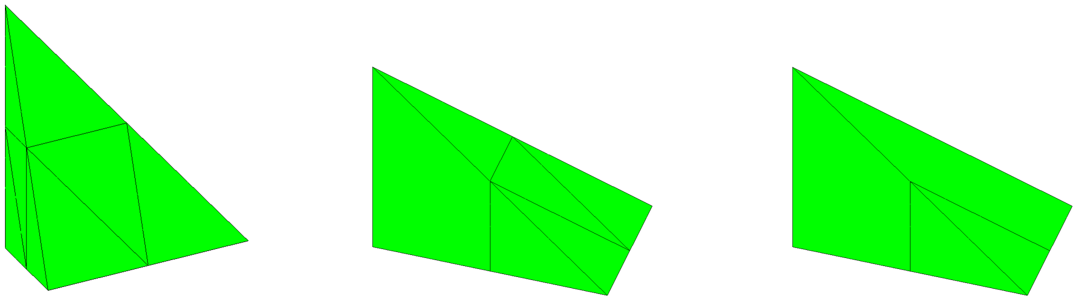
Let us briefly describe the isotropic refinement of the tetrahedron, see Figure 2.14(a). As a first step, we identify four subtetrahedra associated with the vertices and the midedge points of the parent element. By removing those four sub-tetrahedra, an interior octahedron is revealed, see Figure 2.14(b). A tetrahedral



(a) Prism, isotropic

(b) Prism, anisotropic

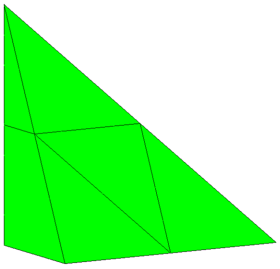
(c) Prism, anisotropic



(d) Tetrahedron, isotropic

(e) Tetrahedron, anisotropic

(f) Tetrahedron, anisotropic



(g) Pyramid, isotropic

Figure 2.13: Supported prism, tetrahedron, and pyramid refinements.

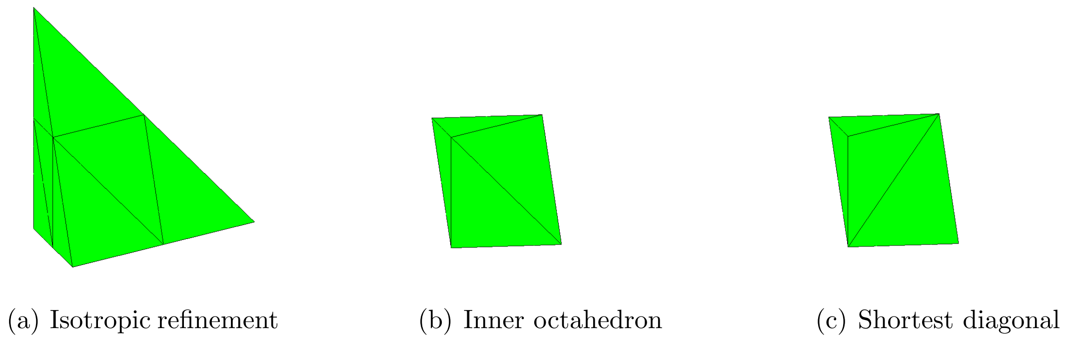

(a) Isotropic refinement (b) Inner octahedron (c) Shortest diagonal

Figure 2.14: Isotropic tetrahedron refinement.

subdiscretization is easily realized by selecting a diagonal of the octahedron, see Figure 2.14(c). Resulting subtetrahedral pairs are congruent reflections. The original reference element is thereby partitioned to four similar (vertex) subtetrahedra and four further (interior) subtetrahedra. While the latter four tetrahedra are no longer similar to the parent element, all 8 subtetrahedra have volume equal to 1/8 of the original element. This is one of the most common refinement schemes for tetrahedra, see, *e.g.*, [13].

As anticipated in the previous section, an **element is broken** by generating son-nodes for the element edge nodes, face nodes and middle node. Once a middle node refinement kind is selected, it implies appropriate refinement kinds for the edges and faces. In fact, things are more complicated, since our element breaking procedure must handle elements with already broken edges and faces. We only discuss the overall logic logic of the element breaking procedure, since the detailed procedure would look cumbersome and fail to communicate the general approach.

We will call an existing face refinement *upgradable* to a given face refinement if there exist refinements of the face son-nodes, called *complementary* refinements, such that the resulting refinement is topologically equivalent to the given one. For example, a horizontally refined quadrilateral face is upgraded to an isotropically refined face by anisotropically refining the two face son nodes, and breaking the edge son node. While an anisotropic refinement of a quadrilateral face can be upgraded to an isotropic one, this is not the case for a triangular face, since the refinement of a quad into three triangles is not implemented in the code.

In simple terms, the element breaking procedure is as follows:

Step 1. Determine face and edge refinements implied by the (element) middle node refinement kind.

Step 2. For each edge node, if the node is not broken and the edge must be refined, generate inactive son nodes.

Step 3. For each face node, if the face must be refined, determine complimentary refinements. If no complimentary refinements need to be applied, just activate the face son nodes. Otherwise, triangular and quadrilateral faces need to be analyzed separately. For a triangular face use the implied face refinement, since at the current statues of development no upgrading is possible, and break it into inactive sons. In the case of a quadrilateral face, if the face is unrefined, break it into inactive sons according to the implied face refinement. Otherwise, i.e., in the case of an already refined quadrilateral face, activate the face son nodes, and break each son node into inactive grandson nodes according to the corresponding complimentary refinement.

Step 4. Break the middle node into active son nodes according to the element refinement flag.

The fundamental assumption of the element breaking procedure is that existing face refinements are upgradable to the implied face refinements. The son nodes that are generated are either active or inactive. Newly generated sons of edge nodes and face nodes are always inactive, while sons of middle nodes are always active. In terms to be made more precise, this is consistent with the fact that constraints can only propagate through edges and faces. The procedure we described above breaks the element without accounting for the status (active/inactive) of its nodes or, in other words, it does not account for constraints. If we were to use the break routine for refining elements, we would end up with multiple levels of constrained nodes, namely inactive nodes whose fathers, grandfathers, great grandfathers and so on are inactive nodes. While this is a perfectly legitimate grid under a topological point of view, it is absolutely impractical for performing computations, since keeping track of the constraints is quite complicated.

In order to keep the propagation of constraints under control, we introduce another procedure, called **element refinement**. Such a procedure, depending upon the status of the face nodes, will either break the element, or break neighboring elements and then the element itself, in order to guarantee mesh regularity across faces, see Section 2.7 for the exact definition of mesh regularity. This is achieved through a stack (first in/last out) algorithm.

```
Data: an element, an element refinement kind
place element on the stack
repeat
    // Step 1:  build a list of refinements to activate the
       faces
1   pick element on top of the stack
    forall element faces do
        if face is active then
            record existing face refinement
        else
            identify neighbor across face
            identify a refinement of the neighbor to activate the face
            add neighboring element to the stack, goto 1
        end
    end
    // Step 2:  refine element on top of the stack
    pick element on top of the stack
    upgrade the refinement kind to accommodate existing face
    refinements
    break element and dequeue it from the stack
until stack is empty
```

Algorithm 2.6.1: Element refinement

Element refinement is governed by two fundamental rules:

Rule 1. An element cannot be refined unless *all* of its face nodes are active.

Rule 2. The middle node refinement kind is *upgraded* to match existing face refinements.

The refinement scheme is a two-step stack algorithm. In the first step, a list of neighboring elements and appropriate refinement kinds, i.e., the stack, is built in order to activate the inactive faces. This allows us to enforce Rule 1. In the second step, the element on the top of the stack is dequeued, its refinement kind is upgraded to match existing face refinements, i.e., Rule 2, and the element is broken using the element breaking procedure we just discussed. The main difficulty arises from the fact that some nodes are potentially activated during the breaking procedure. Hence, before the next element on the shelf can be refined, we need to reanalyze its face nodes status, i.e., repeat the first step. As a result, the two steps are embedded in an infinite loop that terminates when the stack is empty.

The refinement procedure is described in details in Algorithm 2.6.1. We discuss it here in more loose terms.

Step 1. We put the element on a stack. We pick the element on top of the stack and loop over its faces. If the face is active, we determine the face refinement implied by the element refinement and record it. Otherwise, we identify a neighbor across the inactive face and a suitable refinement to activate the face, we add it to the stack, and go back to analyze the last element on the stack, namely the neighboring element we just added. This step terminates when an element with all active faces is found.

Step 2. We pick the element on the top of stack, use the informations collected during the previous step to upgrade its refinement flag to match existing face refinements, break it and dequeue it from the stack.

In the stack algorithm, only constraints that propagates across faces are considered. This fact has a double benefit: it requires a neighbor-across-face routine only—developing a neighbor-across-edge routine is much more complicated—and, *we conjecture*, it prevents *deadlocks*. The weakness of any stack algorithm is that it may require to put on the stack an element that is already on it. This vicious circle is usually called *deadlock*. The subtlety of the problem is that it is not *a priori* clear whether a refinement algorithm is deadlock-free. Most of the times deadlocks emerge after a long trial-and-error process, namely after a code has been successfully employed for some time, it will run into a deadlock. We are not aware of any proof, or disproof, that a given scheme will deadlock or not. On the other hand, we know a counter example involving hexahedra, where the deadlock is caused by constraints that propagate through edges. This insight led us to implement a scheme that relies on constraints through faces only. We have tested our algorithm by refining elements at random and, so far, it never encountered a deadlock.

2.7 Mesh Reconciliation & Mesh Regularity

The goal of mesh reconciliation is to eliminate, by performing additional refinements, double constrained nodes—nodes constrained by a constrained node—, that may have been introduced by refining an element. Only double constraints that originated by edge breaking need to be resolved, since double constraints cannot propagate through face breaking, as a consequence of Rule 1 of the element refinement procedure.

Mesh reconciliation is a complicated process that is performed by embedding in an infinite loop three steps: *(i)* activation of unconstrained nodes; *(ii)* marking of double constrained nodes; *(iii)* additional element refinements. The return con-

```
do
    // Step 1: activate unconstrained nodes
    forall nodes do lower visitation flag
    forall active elements do raise visitation flag for all edges and faces
    forall nodes do
        if node is active then cycle
        if father was not visited then active node
    end
    // Step 2: mark double-constrained nodes
    forall nodes do lower visitation flag
    forall active elements do
        forall element nodes do
            if node is constrained then
                identify the constraining nodes and the type of constraint
                forall constraining nodes do
                    if node is not active then raise visitation flag for all of
                    its ancestors
                end
            end
        end
    end
    // Step 3: perform additional refinements
    initiate list to empty
    forall active elements do
        if an edge of face was visited then
            determine proper refinement, add element to list
        end
    end
    if list is empty then return
    forall elements in the list do
        if the element is unrefined then refine the element
    end
end
```

Algorithm 2.7.1: Mesh reconciliation.

dition is contained in the third step. The whole process relies on the visitation flag. Although this flag plays a role similar to the one of the activation flag, it is merely a service flag. The whole process is detailed in Algorithm 2.7.1. By *active elements* we simply mean the middle nodes which are leaves of the nodal tree.

In rather general terms, the three steps can be described as follows.

Step 1. We lower the visitation flag for all nodes. Then, we loop over all active elements and raise the visitation flag for their edges and faces. Finally, we loop over all inactive nodes and activate those nodes whose father was not visited.

Step 2. We lower the visitation flag for all nodes. Then, for each active element, we examine its nodes. If the node is constrained, then we look at the constraining nodes. For each constraining nodes, if the node is inactive, i.e., constrained, we raise the visitation flag for all of its ancestors. By doing so, we are able to identify double constrained nodes through the visitation flag. This is performed in the following step.

Step 3. We loop over all active elements. If either an edge or a face was visited, then the element needs to be refined. The appropriate refinement kind is determined by examining the existing refinement of its edges and faces and the element is placed on a list of elements to be refined. After the list is created, we examine it. If it is empty, mesh reconciliation has been achieved, otherwise we refine each element on the list that is not refined.

The implementation of the second step is the most critical, since double-constrained nodes need to be identified. In fact, rather than double constraints, we want to to detect constraints that we cannot—or do not want to—handle, and need to be eliminated through additional refinements. Indeed we are giving our own definition of 1-irregular mesh by allowing or forbidding certain scenarios. For example, in our implementation we are allowing a few double constraints, namely some constraining nodes are themselves constrained, hence, strictly speaking, we are not supporting just 1-irregular grids. In any case, since there is not precise definition of 1-irregular mesh in 3D, we do not argue any further. Detecting forbidden double constraints is one of the most complicated parts of code and we believe it will better belong to a software manual rather than a Ph.D. thesis. We will just say that in order to successfully implement those concepts, for each element we need to build a database of constraints using the history of refinements.

2.8 Unassembled Hyper-Matrix Solver

We follow the discussion in [64]. Multi-physics problems usually generate huge linear systems of equations, which are not well conditioned, thus the applicability of iterative solvers is frequently limited. In addition, iterative solvers typically exhibit lack of robustness (indefinite problems, high-contrast materials, elongated elements, etc.). Moreover, iterative solvers may be slower than direct solvers when a system of equations with several right hand sides needs to be solved, as it occurs in the problem of interest, which is solved for multiple incident directions. For all of these reasons we have decided to use a direct solver.

The large size of coupled multiphysics problems typically requires the use of parallel solvers. The current state-of-the-art for direct solvers is *parallel multi-frontal* solvers. The structure of the matrix is analyzed to determine an ordering that will preserve sparsity in the factors in the LU decomposition. This analysis phase produces both an ordering and an assembly tree, which is then used to drive the subsequent numerical factorization and solution phases. At each node of the tree, a dense submatrix (called a frontal matrix) is assembled using data from the original matrix and from the children of the node. The resulting factors are stored for use in the solution phase, and the Schur complement is passed to the parent node for assembly at that node. In the numerical factorization phase, the tree is processed from the leaf nodes to the root. The subsequent forward and backward substitutions during the solution phase process the tree from the leaves to the root and from the root to the leaves, respectively. A crucial aspect is that the assembly tree defines only a partial order for the factorization since the only requirement is that a child must complete its elimination operations before the parent can be fully processed. It is this freedom that enables us to exploit parallelism in the tree (tree parallelism).

The Multi-frontal Massively Parallel Sparse Direct Solver (MUMPS), see [3], has gained vast popularity over the past few years. In the early stages of development, we interfaced our FE library to MUMPS. Yet, the use of an external solver manifests a number of limitations arising from the fact that all insight into the structure of the application is lost.

The shortcomings of the black-box use of a solver are most evident from the workflow of typical *hp*-adaptive FE computations.

Step 1. An initial mesh of the domain of interest is generated.

Step 2. The global stiffness matrix and load vector are computed and stored either in an assembled sparse format or as unassembled element contributions.

Step 3. The sparse linear system in solved via a standard solver package.

Step 4. *hp*-refinements are performed based on an *a posteriori* error estimator.

Step 5. Steps 2-4 are repeated until a stopping criterion is met.

The most clear limitation of this approach is found in Step 3, since the solver ignores what is the relationship, i.e., the history of refinements, between the input data in successive invocations. Furthermore, this information is partially and laboriously reconstructed by graph partitioners, e.g., METIS, used in sparse solver packages.

This motivated the development of a solver specifically tailored to *hp*-finite element applications by a collaborator, Kyungjoo Kim, at The University of Texas at Austin, see [7]. The key feature of such solver is its data structure, called Unassembled Hyper-Matrix (UHM). It preserves application-dependent information which enhances the efficiency of the solver.

The UHM solver exploits the opportunity for a factorization strategy that proceeds by updating the factorization from the previous *hp*-step. The corner stone for the scheme is the UHM, namely storing the matrix as unassembled elemental matrices, hierarchically ordered to mirror the refinement history of the domain. Thus, the interplay between the data structure of the FE library and the one of the solver becomes evident. The factorization the UHM proceeds in terms of elemental matrices by assembling nodes only when they need to be eliminated.

This approach has a number of benefits. In first place, the UHM addresses the issue of traditional matrix storage, which is not flexible enough to easily accommodate rows insertion, a key feature between successive *hp*-steps. Secondly, only the factorization of newly created elemental matrices needs to be computed. Finally, by having access to the refinement history, the UHM solver easily identifies *super nodes*, namely blocks of rows or columns exhibiting similar sparsity patterns, which naturally arise from *hp*-finite elements. Super nodes are of fundamental importance because they can be tackled with dense block algorithms which employ Level 3 (matrix-matrix) BLAS operations, the path to high performance linear algebra.

Chapter 3

The Head Problem

3.1 Problem Set-Up

The domain Ω in which the problem is defined is the interior of a ball, which includes *a* model of the human head, made up of both an acoustic part Ω_a, and an elastic part Ω_e. As anticipated, we have used an idealized head model made up of algebraic surfaces such as spheres, cones, cylinders and planes, see Figure 3.1. A comparison between our idealized model of the middle ear and its real anatomy is illustrated in Figure 3.2. Despite the model being primitive, its main dimensions match the real values that are found in the literature:

skull thickness: $5\,\mathrm{mm}$;
ear canal depth: $25\,\mathrm{mm}$;
cochlea depth: $8\,\mathrm{mm}$;
tympanic membrane thickness: $100\,\mu\mathrm{m}$;
oval and round window thickness: $100\,\mu\mathrm{m}$;
basilar membrane thickness: $5\,\mu\mathrm{m}$.

The model is composed of roughly 20000 blocks which include tetrahedra, prisms, and pyramids as well. The pyramids are a result of the extrusion process, as described in Section 2.1.

We searched the literature in order to determine material constants, namely Young's modulus, Poisson's ratio and density for the elastic media, and density and sound speed for the acoustic fluids. As expected, the values we found differ among them even by one order of magnitude, see Table 3.1. Furthermore, the value of the Poisson ratio for the brain approaches incompressibility, i.e., $\nu = 0.5$. In this case, shear waves (secondary waves) rather then compressional waves (primary waves) are observed. Shear waves propagate with speed $c_s = E(1-\nu)/[2\varrho(1+\nu)]$, which is smaller than the speed of compressional waves $c_p = E(1-\nu)/[\varrho(1+\nu)(1-2\nu)]$. Thus, in the brain the relevant wave length is $\lambda = c_s/f$. For example, if we pick $E = 0.123\,\mathrm{MPa}$, $\nu = 0.48$ and $f = 2000\,\mathrm{Hz}$, which is well within the 20–20000 Hz range of audible frequencies, the wave length for the shear waves is roughly 3 mm. Thus, unless we use an extremely fine mesh in the brain, we cannot capture the behavior of the solution. The work-around that we employed is to model the brain as an acoustical fluid, rather then an elastic medium. We used the material constants for water. The cochlea is supposed to be filled with water as well. The acoustic part Ω_a includes:

(i) air surrounding the head model, bounded by the model outer surface and a truncating sphere, including a portion of the duct leading to the middle ear;

(ii) air in the ear canal between the tympanic membrane and the oval and round window;

(iii) cochlea;

(iv) brain;

(v) an additional layer of air bounded by the truncating sphere and the outer sphere terminating the computational domain, where the equations of acoustics are replaced with the corresponding Perfectly Matched Layer (PML) modification.

Similarly, the elastic part Ω_e of the domain includes:

(i) skull;

(ii) ossicles;

(iii) tympanic membrane, basilar membrane, oval window, and round window.

Only a few of the different types of boundary conditions discussed in Section 1.3 are actually present in the model. Along with interface conditions between the elastic and acoustic subdomains, we impose a homogeneous Dirichlet condition for elasticity on the bottom part of the head model. The Dirichlet condition is necessary to eliminate the rotational-invariant zero energy modes of the system, see [44]. We indeed experienced those modes in our early numerical experiments, when the head was floating in the air, rather then been mounted on its bottom part. As anticipated, the problem is driven through an incident time-harmonic plane pressure wave $p^{\text{inc}} = \hat{p}^{\text{inc}} e^{i\omega t}$. Since we will be computing the space-dependent factor $\hat{p}^{\text{inc}}$ only, we drop the hat for succinctness. Thus, the total pressure is equal to $p + p^{\text{inc}}$, where p is now the acoustic pressure. Notice that in the air surrounding the head model, the relevant physical quantity is the *total* pressure, and the acoustic pressure can be interpreted as a scattered pressure. On the other hand, in the acoustic fluids that

	E [MPa]	ν	ϱ [kg/m^3]	λ [m] (wave length)
Brain				
Ganpule et al. '09, see [38]	0.123	0.499989	1040	
Schiavone et al. '09, see [69]	0.0006	0.4		
	0.180	0.499		
Skull				
Ganpule et al. '09, see [38]	5370	0.19	1710	0.028
Chafi et al. '09, see [14]	15000	0.21	1800	0.045
O'Brien et al. '05, see [48]	6500	0.22	1412	0.033
Johnson, Young '05, see [47]	1600	0.3	(1412)	0.016
Tympanic Membrane				
Gan et al. '06, see [37]	20	0.3	1200	0.065
Malleus				
Gan et al. '06, see [37]	14100	0.3	3590	0.991
Incus				
Gan et al. '06, see [37]	14100	0.3	3230	1.045
Stapes				
Gan et al. '06, see [37]	14100	0.3	2200	1.266
Manubrium				
Gan et al. '06, see [37]	4700	0.3	1000	1.084
Oval Window				
Gan et al. '07, see [36]	5.5	0.3	1200	0.034
Round Window				
Gan et al. '07, see [36]	0.35	0.3	1200	0.009
Basilar Membrane				
Gan et al. '07, see [36]	(base) 50 (mid) 15 (apex) 3	0.3	1200	0.102

Table 3.1: Material constants. The wave length refers to $f = 2000\,\text{Hz}$.

fill the head model, the relevant physical quantity is the *acoustic* pressure, that can be interpreted as a transmitted pressure.

The incident pressure wave induces non-homogeneous interface conditions on the outer boundary of the head model for both the acoustic and the elastic problems. Indeed, using the constitutive equation for elasticity and the interface condition (1.3.8) we obtain

$$(E_{ijkl}\, u_{k,l})n_j = \sigma_{ij}\, n_j = -\big(p + p^{\mathrm{inc}}\big)n_i = g_i - p\, n_i\ ,$$

where we set $g_i = -p^{\mathrm{inc}}\, n_i$. The load on the head model is coming from the total pressure (incident pressure plus acoustic or, in this case, scattered pressure) in the air surrounding it. Similarly, recalling the momentum equation for acoustics and the interface condition (1.3.7), we have that

$$\big(\nabla p + \nabla p^{\mathrm{inc}}\big) \cdot \boldsymbol{n} = -i\omega\varrho_f \boldsymbol{v} \cdot \boldsymbol{n} = \omega^2 \varrho_f \boldsymbol{u} \cdot \boldsymbol{n}\ ,$$

which, by setting $g = -\nabla p^{\mathrm{inc}} \cdot \boldsymbol{n}$, is equivalent to $\nabla p \cdot \boldsymbol{n} = g + \omega^2 \varrho_f \boldsymbol{u} \cdot \boldsymbol{n}$. Thus, the load for the acoustic problem in the outer air is coming from the incident acoustic pressure and the elastic displacement of the adjacent elastic subdomains. Homogeneous conditions, i.e., $g = \boldsymbol{g} = 0$, are imposed on all other acoustics/elasticity interfaces present in the model.

The final formulation of the problem has the form (1.3.9), with the bilinear and linear forms (1.3.10) that simplify as follows

$$b_{ee}(\boldsymbol{u},\boldsymbol{v}) = \int_{\Omega_e} \big(E_{ijkl}\, u_{k,l}\, v_{i,j} - \varrho_s \omega^2\, u_i\, v_i \big) \,, \tag{3.1.1a}$$

$$b_{ae}(p,\boldsymbol{v}) = \int_{\Gamma_I} p\, v_n \,, \tag{3.1.1b}$$

$$b_{ea}(\boldsymbol{u},q) = -\,\omega^2 \varrho_f \int_{\Gamma_I} u_n\, q \,, \tag{3.1.1c}$$

$$b_{aa}(p,q) = \int_{\Omega_a} \big(\nabla p \cdot \nabla q - k^2\, p\, q \big) \,, \tag{3.1.1d}$$

$$l_e(\boldsymbol{v}) = -\int_{\Gamma_{N,e}} p^{\mathrm{inc}}\, v_n \,, \tag{3.1.1e}$$

$$l_a(q) = -\int_{\Gamma_{N,a}} \frac{\partial p^{\mathrm{inc}}}{\partial \boldsymbol{n}}\, q \,, \tag{3.1.1f}$$

where $\boldsymbol{v}$ is a test function relative to the elastic displacement, and should not be confused with the velocity. As anticipated in Section 1.3, a symmetric formulation is readily obtained by dividing the variational formulation for acoustics by factor $\omega^2 \varrho_f$. Therefore bilinear forms $b_{aa}(\cdot\,,\cdot)$, $b_{ea}(\cdot\,,\cdot)$ and linear form $l_a(\cdot)$ become:

$$\begin{aligned} b_{aa}(p,q) &= \frac{1}{\omega^2 \varrho_f} \int_{\Omega_a} \big(\nabla p \cdot \nabla q - k^2\, p\, q \big) \,, \\ b_{ea}(\boldsymbol{u},q) &= -\int_{\Gamma_I} u_n\, q \,, \\ l_a(q) &= -\,\frac{1}{\omega^2 \varrho_f} \int_{\Gamma_{N,a}} \frac{\partial p^{\mathrm{inc}}}{\partial \boldsymbol{n}}\, q \,. \end{aligned}$$

The formulation is indeed symmetric since we always refer to the normal vector *locally*, namely vector $\boldsymbol{n}$ in form $b_{ae}(\cdot\,,\cdot)$ is equal to $-\boldsymbol{n}$ in form $b_{ea}(\cdot\,,\cdot)$.

We give a brief overview of the problem implementation in `hp3d`. Once the geometry has been defined, we set up the physics and the order of approximation for every initial mesh element. In practice, this is achieved by providing a problem-dependent `set initial mesh` subroutine. We choose $p = 2$ as the default order of approximation, and increase it to $p = 4$ in the membranes and to $p = 5$ in the shells, see Figure 3.3, in order to avoid *locking*.

The bulk of the implementation is to provide a subroutine, `elem`, that computes the element stiffness matrix and load vector. Although we are dealing with a coupled problem, the elements are in fact single-physics, namely an element belongs to either Ω_a, where the quantity of interest in the *interior* of the element is the acoustic pressure p, or to Ω_e, where the quantity of interest in the *interior* of the element is the elastic displacement $\boldsymbol{u}$. This suggests to write a subroutine for elastic elements, and a subroutine for acoustical elements. For example, in the case of an elastic element, the blocks corresponding to $b_{aa}(\cdot\,,\cdot)$ and $l_a(\cdot)$ in the formulation (1.3.9) are not present. Further more, by exploiting the symmetry of the formulation, there is not need to compute the block associated to $b_{ea}(\cdot\,,\cdot)$. Thus, we only need to compute the blocks corresponding to $b_{ee}(\cdot\,,\cdot)$, $b_{ae}(\cdot\,,\cdot)$, and $l_e(\cdot)$. Finally, in the case of a *purely* elastic element, namely an elastic element surrounded by elastic elements, the block associated to $b_{ae}(\cdot\,,\cdot)$ is absent as well.

Those considerations suggest the following implementation strategy. We start by noticing that it is essential to group together volume contributions, namely $b_{ee}(\cdot\,,\cdot)$, and surface contributions, namely $b_{ae}(\cdot\,,\cdot)$, $l_e(\cdot)$. For purely psychological reasons—volume integrals are easier to compute than surface integrals!—we deal with volume contributions first and with surface contributions last. Our strategy is as follows: *(i)* compute $b_{ee}(\cdot\,,\cdot)$; loop over element faces, *(ii)* if a Neumann load is applied to the face, compute $l_e(\cdot)$; *(iii)* if the face is on the acoustics/elasticity interface, compute $b_{ae}(\cdot\,,\cdot)$. Thus, it is natural to identify faces as:

- Dirichlet face;
- Purely Neumann face;
- Acoustics/Elasticity face;
- Acoustics/Elasticity face with Neumann load.

A similar discussion holds for acoustics elements.

Finally, since the FE library does not support automatic *hp*-adaptivity, the user has to design a strategy to drive *h*- adaptivity. We developed and implemented an explicit *a posteriori* error estimator for the acoustics/elasticiy coupled problem, see Section 3.4, to drive adaptive refinements.

3.2 Non-dimentionalization

This is a purely mechanical problem and we can choose three independent units for setting up a non-dimensional version of the equations.

Reference length. Parameter h_0 represents the length of a block (tetrahedron, prism or pyramid) in the geometry model necessary to resolve *geometrical* details; we set $h_0 = 10^{-2}$ [m]. Such a parameter should not be confused with the finite element length that varies accordingly to wave number regime and order of approximation. The choice of reference length implies that the geometry data for the head problem (non-dimensional nodal coordinates) will be of order 1-10.

Reference angular frequency. The reference angular frequency ω_0 corresponds to the dominating frequency in the spectrum of the actual sound source.

Reference pressure. We set $p_0 = 100$ [Pa], a value corresponding to the threshold of pain for the human ear in air.

We define now the following non-dimensional quantities:

Coordinates: $\hat{x} = x_i/h_0$,

Angular frequency: $\hat{\omega} = \omega/\omega_0$,

Incident pressure: $\hat{p}^{\mathrm{inc}} = p^{\mathrm{inc}}/p_0$.

This implies that $\partial/\partial x_i = (1/h_0)\,(\partial/\partial\hat{x}_i)$ and $dx_i = h_0\,d\hat{x}_i$, hence the variational form for elasticity can be equivalently written as:

$$\int_{\Omega_e}\left(E_{ijkl}\,\frac{\partial u_k}{\partial x_l}\,\frac{\partial v_i}{\partial x_j} - \varrho_s\,\omega^2\,u_i v_i\right) + \int_{\Gamma_I} p\,v_n = -\int_{\Gamma_{N,e}} p^{\mathrm{inc}}\,v_n\ ,$$

$$\int_{\Omega_e}\left(E_{ijkl}\,\frac{\partial(u_k/h_0)}{\partial\hat{x}_l}\,\frac{\partial(v_i/h_0)}{\partial\hat{x}_j} - \varrho_s\omega_0^2\hat{\omega}^2\,u_i\,v_i\right)h_0^3 + \int_{\Gamma_I} p\,v_n\,h_0^2 = -\int_{\Gamma_{N,e}} p_0\hat{p}^{\mathrm{inc}}\,v_n\,h_0^2\ ,$$

$$\int_{\Omega_e}\left(\frac{E_{ijkl}}{p_0}\,\frac{\partial(u_k/h_0)}{\partial\hat{x}_l}\,\frac{\partial(v_i/h_0)}{\partial\hat{x}_j} - \frac{\varrho_s\omega_0^2 h_0^2}{p_0}\,\hat{\omega}^2\,\frac{u_i}{h_0}\,\frac{v_i}{h_0}\right) + \int_{\Gamma_I}\frac{p}{p_0}\,\frac{v_n}{h_0} = -\int_{\Gamma_{N,e}}\hat{p}^{\mathrm{inc}}\,\frac{v_n}{h_0}\ .$$

A similar reasoning applies to the variational form for acoustics. This leads us to define the following quantities that are indeed non-dimensional:

Elastic displacements: $\hat{u}_i = u_i / h_0$,

Scattered pressure: $\hat{p} = p / p_0$,

Elastic module (for each elastic material): $\hat{E} = E / p_0$,

Densities (for each acoustic/elastic material): $\hat{\varrho} = \big(\varrho\, \omega_0^2\, h_0^2\big) / p_0$,

Wave speeds (for each acoustic/elastic material): $\hat{c} = c / \big(\omega_0\, h_0\big)$.

In practice, the form of the equations remains identical to the original formulation, with the non-dimensional quantities replacing the original ones. As usual, we drop the karats in the notation.

The weak coupling offers the possibility of an additional scaling of elastic displacements. This is possible because there are no strong continuity conditions relating pressure and elastic displacements across the fluid/solid interfaces. We replace u_i by $t\, u_i$, where t is some scaling parameter, in the variational formulation:

$$
\begin{aligned}
t\, b_{ee}(\boldsymbol{u}, \boldsymbol{v}) + b_{ae}(p, \boldsymbol{v}) &= l_e(\boldsymbol{v}) \qquad \forall\, \boldsymbol{v} \in U \,, \\
t\, b_{ea}(\boldsymbol{u}, q) + b_{aa}(p, q) &= l_a(q) \qquad \forall\, q \in V \,.
\end{aligned}
$$

Again, to preserve the symmetry of the formulation, we divide the second equation by the scaling factor t to obtain the final form of the equations.

$$
\begin{aligned}
t\, b_{ee}(\boldsymbol{u}, \boldsymbol{v}) + b_{ae}(p, \boldsymbol{v}) &= l_e(\boldsymbol{v}) \qquad \forall\, \boldsymbol{v} \in U \,, \\
b_{ea}(\boldsymbol{u}, q) + \frac{1}{t}\, b_{aa}(p, q) &= \frac{1}{t}\, l_a(q) \qquad \forall\, q \in V \,.
\end{aligned}
$$

In order to improve the conditioning and minimize the round-off effects, we choose the scaling parameter in such a way that, after the scaling, the stiffness terms in elasticity (for the skull) and acoustics (for the air) are of the same order, i.e.

$$tE = \frac{1}{t\varrho_f \omega^2} \quad \Rightarrow \quad t = \frac{1}{\sqrt{E\varrho_f}\omega} \, .$$

Notice that as a byproduct of the optimal scaling, the rescaled elastic displacement is expected to be of order one. Indeed, with the new stiffness terms being of the same order, the rescaled elastic displacements and pressure should be of the same order. As the intensity of the scattered pressure is expected to be of order of incident pressure, the non-dimensional scattered pressure should be of order one and, therefore, the same should hold for the rescaled elastic displacements.

In the PML part of the acoustical domain, the bilinear form b_{aa} is modified as follows:

$$\begin{aligned} b_{aa}(p,q) = \int_{\Omega_{a,\mathrm{PML}}} \Big(& \frac{z^2}{z' r^2} \frac{\partial p}{\partial r} \frac{\partial q}{\partial r} + \frac{z'}{r^2} \frac{\partial p}{\partial \psi} \frac{\partial q}{\partial \psi} + \frac{z'}{r^2 \sin^2 \psi} \frac{\partial p}{\partial \theta} \frac{\partial q}{\partial \theta} + \\ & - \frac{z' z^2}{r^2} kpq \Big) r^2 \sin\psi dr d\psi d\theta \, . \end{aligned}$$

Here r, ψ, θ denote the standard spherical coordinates and $z = z(r)$ is the PML stretching factor defined as follows,

$$z(r) = \left(1 - \frac{i}{k} \left[\frac{r-a}{b-a} \right]^{\alpha} \right) r \, . \tag{3.2.2}$$

Here a is the radius of the truncating sphere, b is the external radius of the computational domain ($b-a$ is thus the thickness of the PML layer), i denotes the imaginary unit, k is the acoustical wave number, and r is the radial coordinate. In computations, all derivatives with respect to spherical coordinates are expressed in terms of the standard derivatives with respect to Cartesian coordinates. In all reported computations, parameter $\alpha = 6$. For a detailed discussion on derivation of PML modifications and effects of higher order discretizations see [58, 25].

3.3 Geometry Convergence Test

As already discussed, the Geometry Modeling Package builds a parameterization $\mathbf{x}_b$ for each block in the mesh and returns its point values and derivatives with respect to the master element's coordinates. This information is used to determine the corresponding isoparametric maps $\mathbf{x}_{hp}$ via PBI. By studying convergence rates of $\|\mathbf{x}_b - \mathbf{x}_{hp}\|_{1,\Omega}$ as refinements are performed, we investigate the regularity of the geometry maps and verify that GMP is working correctly. Since performing multiple global uniform refinements of the actual mesh used for the coupled problem is too expensive, we resort to smaller meshes that recreate the scenarios encountered in the original mesh. Let's recall that PBI is constructed in a way such that it guarantees optimal rates of convergence, namely the following well-known h-estimate holds:

$$\|\mathbf{x}_b - \mathbf{x}_{hp}\|_{1,\Omega} \leq C\, h^{\min\{p,r\}} \|\mathbf{x}_b\|_{r+1,\Omega}\,.$$

Let's estimate h in terms of geometry dof's:

$$\#\,\text{geometry dof's} \;\approx\; p^3\,\#\,\text{elements} \;\approx\; p^3\,\frac{|\Omega|}{h^3} \quad\Rightarrow\quad h \;\approx\; C\,\#\,\text{geometry dof's}^{-1/3}\,,$$

where we lumped the dependency upon p into the constant C. Thus, provided that the geometry map is sufficiently regular, we expect a rate of convergence $\alpha = -p/3$.

Furthermore, PBI can be employed to investigate the regularity of the geometry maps. Indeed, if suboptimal rates of convergence are observed, then poor regularity of the geometry maps is to blame. It is fundamental to remark that low regularity of the geometry maps will spoil the convergence rate of the FE solution.

While we have performed several geometry tests, we discuss only the more significant ones. The first test employs a mesh composed of eight tetrahedra, which occupy an octant of a sphere each, surrounded by eight prisms, see Figure 3.4. The parameterization of the exact geometry map is visible in Figure 3.4(a); it produces a uniform grid on each octant. The goal of this test is to verify that both the tetrahedron and prism parameterizations are behaving as expected. The incremental nature of PBI is of significant help in debugging. Indeed, when $p = 2$, PBI employes the value of the parameterization at the element's vertices, and its derivative on the element edges only; when $p = 3$, the derivative of the parameterization on the element faces are used as well; finally, when $p = 4$, the derivative of the parameterization inside the element is used as well. This fact further emphasizes the crucial role played by the geometry in high order final elements. We report the observed rates of convergence for $p = 2, 3, 4$ in Table 3.2.

It is essential to notice that after a refinement is performed all geometry dof's are updated. This means that, for each active element, we use the composition of the exact geometry map and the refinement map, see Section 2.3, to produce new geometry dof's. Indeed, a clear difference in shape is noticeable between Figure 3.4(a) and Figure 3.4(b). As refinements are performed, see Figure 3.4(c) and 3.4(d), such a difference becomes less and less evident. An even more dramatic effect would be

refinement level	$\|\mathbf{x}_b - \mathbf{x}_{hp}\|_1$	$\|\mathbf{x}_b\|_1$	$\dfrac{\|\mathbf{x}_b - \mathbf{x}_{hp}\|_1}{\|\mathbf{x}_b\|_1}$	α
$p = 2$				
0	0.93321 E+00	0.78420 E+01	0.11900 E+00	
1	0.29517 E+00	0.78420 E+01	0.37640 E−01	−0.581885
2	0.78241 E−01	0.78420 E+01	0.99772 E−02	−0.662017
3	0.19878 E−01	0.78420 E+01	0.25348 E−02	−0.670172
4	0.49901 E−02	0.78420 E+01	0.63633 E−03	−0.669729
$p = 3$				
0	0.22322 E+00	0.78420 E+01	0.28464 E−01	
1	0.37224 E−01	0.78420 E+01	0.47467 E−02	−0.861088
2	0.51431 E−02	0.78420 E+01	0.65584 E−03	−0.969399
3	0.66153 E−03	0.78420 E+01	0.84357 E−04	−0.996336
4	0.83329 E−04	0.78420 E+01	0.10626 E−04	−1.001154
$p = 4$				
0	0.43486 E−01	0.78420 E+01	0.55453 E−02	
1	0.45516 E−02	0.78420 E+01	0.58041 E−03	−1.064168
2	0.32658 E−03	0.78420 E+01	0.41646 E−04	−1.280409
3	0.21321 E−04	0.78420 E+01	0.27189 E−05	−1.321644
4	0.13491 E−05	0.78420 E+01	0.17203 E−06	−1.332122

Table 3.2: Geometry convergence rates for the mesh shown in Figure 3.4.

obtained with linear elements.

The goal of the second geometry test is to reproduce the situation that happens in a membrane when anisotropic refinements of pyramids and tetrahedra are performed in the outer ribbon. We built a mesh composed of a pyramid and a tetrahedron only that mimics what ideally happens during the extrusion process that generates a circular membrane, see Figure 3.5. We report the observed rates of convergence for $p = 2, 3, 4$ in Table 3.3. The fact that the observed rates exceed the predicted rates is explained by the fact that anisotropic rather than isotropic refinements are performed.

3.4 *A posteriori* Error Indicator

A posteriori error estimates aim at answering the natural question of whether a finite element solution is a sufficiently accurate approximation, in some sense to be specified, to the exact solution of the problem of interest. An *a posteriori* error estimator should provide an upper and/or lower bound on the error between the exact and approximate solution. The mathematical theory behind such estimators was started by the pioneering works of Oden et al., see [63], and Babuška and Rheinboldt, see [75], and developed to the point that nowadays an extensive literature on the subject exists. Excellent references are the books by Ainsworth and Oden [2], and by Verfürth [83], as well as the book by Babuška and Strouboulis, e.g., [76], which gives a vast and profound analysis of the subject. Finally we recall the recent work by Vohralík, see, e.g., [46].

An *a posteriori* error estimator is based on the following idea. Let u and u_h

refinement level	$\|\mathbf{x}_b - \mathbf{x}_{hp}\|_1$	$\|\mathbf{x}_b\|_1$	$\dfrac{\|\mathbf{x}_b - \mathbf{x}_{hp}\|_1}{\|\mathbf{x}_b\|_1}$	α
$p = 2$				
0	0.33736 E−01	0.32716 E+01	0.10312 E−01	
1	0.71072 E−02	0.32716 E+01	0.21724 E−02	−1.157093
2	0.16859 E−02	0.32716 E+01	0.51532 E−03	−1.093870
3	0.41571 E−03	0.32716 E+01	0.12707 E−03	−1.050723
4	0.10357 E−03	0.32716 E+01	0.31657 E−04	−1.026968
$p = 3$				
0	0.21712 E−02	0.32716 E+01	0.66365 E−03	
1	0.22569 E−03	0.32716 E+01	0.68985 E−04	−1.515385
2	0.26675 E−04	0.32716 E+01	0.81534 E−05	−1.541817
3	0.32859 E−05	0.32716 E+01	0.10044 E−05	−1.529010
4	0.40924 E−06	0.32716 E+01	0.12509 E−06	−1.517221
$p = 4$				
0	0.11193 E−03	0.32716 E+01	0.34213 E−04	
1	0.57769 E−05	0.32716 E+01	0.17658 E−05	−1.884216
2	0.34057 E−06	0.32716 E+01	0.10410 E−06	−1.987431
3	0.20964 E−07	0.32716 E+01	0.64079 E−08	−2.003921
4	0.13053 E−08	0.32716 E+01	0.39897 E−09	−2.005159

Table 3.3: Geometry convergence rates for the mesh shown in Figure 3.5.

be the solutions of the continuous and discrete problem respectively,

$$b(u,v) = l(v) \qquad \forall\, v \in V \qquad ; \qquad b(u_h, v_h) = l(v_h) \qquad \forall\, v_h \in V_h \; ,$$

where $b(\cdot,\cdot) : V \times V \to \mathbb{F}$ is a bilinear and continuous form, V a Hilbert space on the real (or complex) field $\mathbb{F}$, and l belongs to the dual space V^*. If the approximation is conformal, i.e., $V_h \subset V$, then the error satisfies the residual equation

$$b(u-u_h, v) = b(u,v) - b(u_h, v) = l(v) - b(u_h, v) \qquad \forall\, v \in V \; . \tag{3.4.3}$$

Since u_h is known, the error $e_h := u - u_h$ the right hand side of the last equality is a linear and continuous operator on V, hence the error satisfies the continuous problem with a modified right-hand side. Moreover, the error satisfies the Galerkin orthogonality:

$$b(u-u_h, v_h) = 0 \qquad \forall v_h \in V_h \; . \tag{3.4.4}$$

A posteriori error estimators employ relation (3.4.3) and (3.4.4) to provide information about $\|u - u_h\|$ in some suitable problem-dependent norm. In principle, one could solve problem (3.4.4) for the error $e_h := u - u_h$. However, this problem is in general more computationally expansive then the original problem. Indeed, unless we use a richer discrete space to solve the residual equation (3.4.3), such problem reduces to the Galerkin orthogonality condition (3.4.4), which gives $e_h = 0$ when solved in V_h. In fact, our computational effort would be better invested in obtaining an improved approximation u_h, which we would expect to be somehow closer to the exact solution. Thus, it should be implicitly understood that computing an a posteriori error estimator is cheaper then solving the original problem.

We start our derivation by noticing that the acoustics/elasticity problem can be *decoupled* through the approximation $p \approx p_h$ and $\boldsymbol{u} \approx \boldsymbol{u}_h$. As naive as this procedure could seem, it is indeed quite similar to the approximation of the flux between adjacent elements used by a standard implicit error estimator – more on this to follow. Since the exact solutions have been replaced by their finite element approximations, it is customary to talk about an error indicator, rather than an error estimator. We follow this convention in the remainder of the discussion.

By defining $B_e(\cdot\,,\cdot) = b_{ee}(\cdot\,,\cdot)$ and $L_e(\cdot) = l_e(\cdot) - b_{ae}(p,\cdot)$, the residual equation for the elastic problem can be written as:

$$\begin{aligned} B_e(\boldsymbol{e}_h,\boldsymbol{v}) &= L_e(\boldsymbol{v}) - B_e(\boldsymbol{u}_h,\boldsymbol{v}) \\ &= -\int_{\Gamma_{N,e}} p^{\mathrm{inc}} v_n - \int_{\Gamma_I} p\, v_n - \int_{\Omega_e} \big((E\,\nabla\boldsymbol{u}_h) : \nabla\boldsymbol{v}_h - \varrho_s\omega^2\,\boldsymbol{u}_h\cdot\boldsymbol{v} \big) \\ &= -\int_{\Gamma_{N,e}} p^{\mathrm{inc}} v_n - \int_{\Gamma_I} p\, v_n - \sum_{K\subset\Omega_e}\int_K \big((E\,\nabla\boldsymbol{u}_h) : \nabla\boldsymbol{v}_h - \varrho_s\omega^2\,\boldsymbol{u}_h\cdot\boldsymbol{v} \big)\,. \end{aligned}$$

Integration by parts over each element K in the elastic domain yields:

$$B_e(\boldsymbol{e}_h,\boldsymbol{v}) = \sum_K \int_K \boldsymbol{r}_h\cdot\boldsymbol{v} - \sum_{F_{\mathrm{int}}}\int_F [\![E\,\nabla\boldsymbol{u}_h]\!]\cdot\boldsymbol{v} + \sum_{F_I}\int_F \big(-(p^{\mathrm{inc}}+p)\,\boldsymbol{n} - (E\,\nabla\boldsymbol{u}_h)\,\boldsymbol{n} \big)\cdot\boldsymbol{v}\,,$$

where F_{int} are the faces in the interior of the elastic domain, F_I are the faces sitting on the acoustics/elasticity interface Γ_I, $\boldsymbol{r}_h$ is the element residual, $\boldsymbol{u}_{h,K}$ is the restriction of $\boldsymbol{u}_h$ to K, and $[\![\cdot]\!]$ is the jump over face F adjacent to element K and J:

$$\begin{aligned} \boldsymbol{r}_h &= \varrho_s\omega^2\,\boldsymbol{u}_h + \operatorname{div}(E\,\nabla\boldsymbol{u}_h)\,, \\ [\![E\,\nabla\boldsymbol{u}_h]\!] &= (E\,\nabla\boldsymbol{u}_{h,K})\,\boldsymbol{n}_K + (E\,\nabla\boldsymbol{u}_{h,J})\,\boldsymbol{n}_J\,. \end{aligned}$$

The element normals are always assumed to be pointing outwards. For each elastic element K we define the face residual $\boldsymbol{R}_h$ as:

$$\begin{aligned}
\boldsymbol{R}_h &= -\big[p^{\text{inc}}\,\boldsymbol{n}_K + p\,\boldsymbol{n}_K + (E\,\nabla\boldsymbol{u}_h)\,\boldsymbol{n}_K\big] && \text{on } F = \partial K \cap \Gamma_I\,,\\
\boldsymbol{R}_h &= -\big[(E\,\nabla\boldsymbol{u}_{h,K})\,\boldsymbol{n}_K + (E\,\nabla\boldsymbol{u}_{h,J})\,\boldsymbol{n}_J\big] && \text{on } F = \partial K \cap \partial J\,,\\
\boldsymbol{R}_h &= 0 && \text{on } F = \partial K \cap \Gamma_{D,e}\,.
\end{aligned}$$

For the face residual to be well-defined, we need to trivially extend p^{inc} to the whole interface Γ_I; furthermore, $\boldsymbol{R}_h$ is a computable quantity provided we make the approximation $p \approx p_h$.

We proceed in a standard way to derive an estimate for $\|\boldsymbol{e}_h\|$ from the residual equation. By virtue of the Galerking orthogonality we have that:

$$B_e(\boldsymbol{e}_h, \boldsymbol{v}) = \sum_K \int_K \boldsymbol{r}_h \cdot (\boldsymbol{v} - \Pi_h \boldsymbol{v}) + \sum_F \int_F \boldsymbol{R}_h \cdot (\boldsymbol{v} - \Pi_h \boldsymbol{v})\,,$$

where $\Pi_h : W^{s,p}(\Omega_e) \to U_h$ is *an* approximation operator. Since the solution may not possess significant regularity beyond H^1-smoothness, the operator Π_h we need to employ is more sophisticated than the usual interpolation operator, that in fact requires additional regularity of the function being interpolated. A choice for Π_h is the Clément-type operator constructed by Bernardi and Girault in [5]. The following estimates hold:

$$|u - \Pi_h u|_{W^{t,p}(K)} \leq C h^{s-t} |u|_{W^{s,p}(\tilde{K})}\,, \tag{3.4.5a}$$

$$|u - \Pi_h u|_{W^{t,p}(F)} \leq C h^{s-t-1/p} \|u\|_{W^{s,p}(\tilde{K})}\,, \tag{3.4.5b}$$

where $1 \leq p < \infty$, $0 \leq t \leq 1$ and $t \leq s \leq q+1$ and q is the degree of the finite element space; by $\tilde{K}$ we indicate the patch of elements that share either an edge or a face with element K. The Cauchy-Schwarz inequality and estimates (3.4.5), applied for $s = 1$, $t = 0$, $p = 2$, imply that

$$\begin{aligned}
|B_e(\boldsymbol{e}_h, \boldsymbol{v})| &\leq \sum_K \|\boldsymbol{r}_h\|_{0,K} \|\boldsymbol{v} - \Pi_h \boldsymbol{v}\|_{0,K} + \sum_F \|\boldsymbol{R}_h\|_{0,F} \|\boldsymbol{v} - \Pi_h \boldsymbol{v}\|_{0,F} \\
&\leq C \|\boldsymbol{v}\|_{1,\Omega_e} \bigg(\sum_K h \, \|\boldsymbol{r}_h\|_{0,K} + \sum_F h^{1/2} \, \|\boldsymbol{R}_h\|_{0,F} \bigg) \\
&\leq C \|\boldsymbol{v}\|_{1,\Omega_e} \bigg(\sum_K h^2 \, \|\boldsymbol{r}_h\|_{0,K}^2 + \sum_F h \, \|\boldsymbol{R}_h\|_{0,F}^2 \bigg)^{1/2} ,
\end{aligned}$$

where the last inequality is a consequence of the discrete inequality $\sum_{i=1}^n a_i \leq n^{1/2} \big(\sum_{i=1}^n a_i^2 \big)^{1/2}$. Form $B_e(\cdot \, , \cdot)$ is the sum of a coercive term and a compact perturbation, thus an inf-sup condition holds

$$\exists \, \gamma_e > 0 \quad \text{such that} \qquad \sup_{\{\boldsymbol{v} \in U \, : \, \boldsymbol{v} \neq 0\}} \frac{|B_e(\boldsymbol{e}_h, \boldsymbol{v})|}{\|\boldsymbol{v}\|} \geq \gamma_e \|\boldsymbol{e}_h\| \, .$$

This implies that

$$\begin{aligned}
\|\boldsymbol{e}_h\|_{1,\Omega_e}^2 &\leq C \bigg(\sum_K h^2 \, \|\boldsymbol{r}_h\|_{0,K}^2 + \sum_F h \, \|\boldsymbol{R}_h\|_{0,F}^2 \bigg) \\
&\leq C \sum_K \left(h^2 \, \|\boldsymbol{r}_h\|_{0,K}^2 + \frac{1}{2} h \, \|\boldsymbol{R}_h\|_{0,\partial K}^2 \right) ,
\end{aligned}$$

where, since we are ultimately interested in element contributions, we split the face contributions evenly between neighboring elements.

The quantity

$$\theta_K^2 = h^2 \, \|\boldsymbol{r}_h\|_{0,K}^2 + \tfrac{1}{2} h \, \|\boldsymbol{R}_h\|_{0,\partial K}^2 \tag{3.4.6}$$

is the natural candidate to estimate the error associated to element K. We just proved that $\|\boldsymbol{e}_h\|_{1,\Omega_e}^2 \leq C \sum_K \theta_K^2$; if the reverse inequality holds, then the indicator is said to be *efficient*. This is of practical importance, since it reveals that the behavior of the indicator mimics the behavior of the error as the mesh size decreases. In fact, since our error indicator fits the general framework of explicit error indicators, by applying a standard theory, see for example Section 2.3 of [2], it is possible to prove the following quasi-local estimate:

$$\theta_K^2 \leq C\Big(\|\boldsymbol{e}_h\|^2_{1,\tilde{K}} + h \|\boldsymbol{g} - \bar{\boldsymbol{g}}\|_{0,\partial K \cap \Gamma_I} \Big) \,,$$

where $\boldsymbol{g} = -(p^{\mathrm{inc}}\, \boldsymbol{n} + p\, \boldsymbol{n})$ and $\bar{\boldsymbol{g}}$ is a piecewise continuous approximation of $\boldsymbol{g}$ coming from the finite element space. This bound is of particular relevance for adaptive schemes, since it implies that, if the indicator θ_K is large on a particular element, then the error on K is large as well.

A completely equivalent discussion applies to the case of acoustical elements. In this case, the element and face residual are defined as:

$$\begin{aligned}
\boldsymbol{r}_h &= \Delta\, p_h + k^2 p_h && \text{in } \ K \,, \\
\boldsymbol{R}_h &= -\big(p^{\mathrm{inc}}\, \boldsymbol{n}_K + \omega^2 \varrho_f\, \boldsymbol{u}_h \cdot \boldsymbol{n}_K - \nabla p_h \cdot \boldsymbol{n}_K\big) && \text{on } F = \partial K \cap \Gamma_I \,, \\
\boldsymbol{R}_h &= -\big(\nabla p_{h,K} \cdot \boldsymbol{n}_K + \nabla p_{h,J} \cdot \boldsymbol{n}_J\big) && \text{on } F = \partial K \cap \partial J \,, \\
\boldsymbol{R}_h &= 0 && \text{on } F = \partial K \cap \Gamma_{D,a} \,,
\end{aligned}$$

where, as anticipated, we resorted to the approximation $\boldsymbol{u} \approx \boldsymbol{u}_h$. The element error indicator θ_K is formally equivalent to the one defined in (3.4.6).

Error estimators that are computed directly from the finite element approximation and the data for the problem of interest are referred to as *explicit*. This is indeed the case of the error estimator we just derived. The definition of the element residual involves second order derivatives. If we were to employ such an estimator, second order derivative would have to be computed numerically, since neither GMP nor the shape functions routine supports second order derivatives. Yet, since we are mostly using second order isoparametric elements, numerical computation of second derivatives should not be an issue.

An alternative that avoids the computation of the second order derivatives is an *implicit* estimator. Those estimators require the approximate solution of local auxiliary boundary value problems. We briefly discuss implicit error estimators in order to make a comparison with the explicit error estimator we just developed. We shall follow the approach presented by Oden et al. in [61] and restrict ourselves to the case of the Poisson equation.

Let us consider the Poisson problem on a regular domain Ω with mixed boundary conditions:

$$\begin{aligned} -\Delta u &= f \qquad && \text{in } \Omega \,, \\ \frac{\partial u}{\partial \boldsymbol{n}} &= g \qquad && \text{on } \Gamma_N \,, \\ u &= 0 \qquad && \text{on } \Gamma_D \,, \end{aligned}$$

where $f \in L^2(\Omega)$, and $g \in H^{1/2}(\Gamma_N)$. Let $\mathcal{T}_h$ be a finite element partition of Ω and consider the corresponding finite element space $V_h \subset V := \{v \in H^1(\Omega) \,:\, v|_{\Gamma_D} = 0\}$. The main idea relies on a projection operator onto a space of bubble-like functions. Let $I_h : H^r(\Omega) \to V_h$ be *an* interpolation operator that is constructed element-wise, for example the PBI operator introduced in Section 2.3. For each finite element space V_H, we consider a larger space V_h that is obtained by performing h-refinements and/or p-enrichments. We define a space $M_h \subset V_h$ of bubble-like functions as:

$$M_h = \{ v_h \in V_h \ : \ I_H v_h \equiv 0 \} \, .$$

Because of the local nature of I_h, such a space can be obtained by piecing together element spaces of bubble functions, $M_h(K)$. We define a projection operator on M_h via the bilinear form:

$$P_H : V \to M_h \quad , \quad B(v_h - P_H v_h, w_h) = 0 \qquad \forall \, w_h \in M_h \, .$$

We set $e_h = u - u_h$, $E_H = u_h - u_H$ and let $\| \cdot \|_E$ be the energy norm.

Using standard techniques, it is immediate to show that:

$$B(E_H, v_H) = 0 \qquad \forall \, v_H \in V_H \quad ; \quad \|e_H\|_E = \|e_h\|_E + \|E_H\|_E \, . \tag{3.4.7}$$

We expect $\|e_h\|_E$ to be orders of magnitude smaller than $\|E_H\|_E$, thus we make the approximation approximation $\|e_h\|_E \approx \|E_H\|_E$. Using the estimate $\|v_h - I_H v_h\|_E \leq C \|v_h\|_E$, which is an immediate consequence of the finite dimension of V_h, the first one for relations (3.4.7), and the definition of the projection operator, we have that

$$\|E_H\|_E = \sup_{v_h \in V_h} \frac{|B(E_H, v_h)|}{\|v_h\|_E} = \sup_{v_h \in V_h} \frac{|B(P_H E_H, v_h - I_H v_h)|}{\|v_h\|_E} \leq C \|P_H E_H\|_E \, .$$

Simultaneously, since P_H is a projection, be obtain the converse estimate $\|P_H E_H\|_E \leq \|E_H\|_E$. Thus, the projection $P_H E_H$ on M_h is a good indicator of the error E_H. The Galerking orthogonality implies, in particular, that $e_h \perp_E M_h$. Thus $P_H E_H = P_H e_H$.

In order to estimate $\|P_H e_H\|_E = \|P_H e_H\|_E$, we employ the classical property of orthogonal projections:

$$\begin{aligned}
-\tfrac{1}{2}\|P_H e_H\|_E^2 &= \inf_{v_h\,\in\, M_h} \left[\tfrac{1}{2}B(v_h,v_h) - B(e_H,v_h)\right] \\
&= \inf_{v_h\,\in\, M_h} \sum_K \left[\tfrac{1}{2}B_K(v_h,v_h) + \int_K \Delta e_H v_h - \oint_{\partial K} \frac{\partial u}{\partial \boldsymbol{n}_K} v_h + \oint_{\partial K} \frac{\partial u_H}{\partial \boldsymbol{n}_K} v_h\right] \\
&= \inf_{v_h\,\in\, M_h} \sum_K \left[\tfrac{1}{2}B_K(v_h,v_h) + \int_K \Delta e_H v_h - \int_{\partial K\cap\Gamma_N} g v_h + \right. \\
&\qquad\qquad \left. + \int_{\partial K\setminus\partial\Omega} \frac{\partial u_H}{\partial \boldsymbol{n}_K} v_h + \int_{\partial K\cap\Gamma_N} \frac{\partial u_H}{\partial \boldsymbol{n}_K} v_h\right] \\
&= \inf_{v_h\,\in\, M_h} \sum_K \left[\tfrac{1}{2}B_K(v_h,v_h) + \int_K \Delta e_H v_h + \int_{\partial K\setminus\partial\Omega} \frac{\partial u_H}{\partial \boldsymbol{n}_K} v_h + \right. \\
&\qquad\qquad \left. + \int_{\partial K\cap\Gamma_N} \left(\frac{\partial u_H}{\partial \boldsymbol{n}_K} - g\right) v_h\right],
\end{aligned}$$

where $B_K(\cdot\,,\cdot)$ is the bilinear form over element K, and we used the fact that $\sum_K \oint_{\partial K} \partial u/\partial \boldsymbol{n}_K\, v_h = \int_{\Gamma_N} g\, v_h$.

The next step is to reduce the functional

$$G : V \to \mathbb{R} \quad , \quad G(v_h) = \tfrac{1}{2}B(v_h,v_h) - B(e_H,v_h)$$

to a sum of element contributions. We relax the requirement of interelement continuity by introducing a larger space of bubble-like functions $\mathscr{M}_h = \prod_K M_h(K)$, and reimpose it via a Lagrange multiplier λ defined on $\bigcup_K(\partial K \setminus \partial\Omega)$. We restrict the discussion to the 2D case. We assign a *global* orientation to each side S in the mesh and, for each point $s \in S$, we define the function:

$$\operatorname{sign}(s) = \begin{cases} +1 & S \text{ is oriented clockwise around } \partial K \\ -1 & \text{otherwise} \end{cases} .$$

Thus, the constrained minimization problem for the functional G becomes a saddle-point problem for the Lagrangian

$$L(v_h,\lambda) = G(v_h) - \sum_{\partial K}\oint_{K}\operatorname{sign}(s)\lambda v_h = \sum_{K}\Big[\tfrac{1}{2}B_K(v_h,v_h) + \int_K \Delta e_H v_h + \\ + \int_{\partial K\setminus\partial\Omega}\frac{\partial u_H}{\partial \boldsymbol{n}_K}v_h + \int_{\partial K\cap\Gamma_N}\Big(\frac{\partial u_H}{\partial \boldsymbol{n}_K} - g\Big)v_h - \oint_{\partial K}\operatorname{sign}(s)\lambda v_h\Big]\,.$$

Moreover, because of a result from minmax theory, we can invert the order of the supremum and infimum:

$$-\tfrac{1}{2}\|P_H e_H\|_E^2 = \inf_{v_h\,\in\,\mathscr{M}_h}\sup_{\lambda} L(v_h,\lambda) = \sup_{\lambda}\inf_{v_h\,\in\,\mathscr{M}_h} L(v_h,\lambda)\,.$$

Since $\mathscr{M}_h$ is a broken space, we can proceed element-wise. For each λ, a stationary point $\varphi_K^\lambda \in M_h(K)$ of the Lagrangian satisfies the elemental problem

$$B_K(\varphi_K^\lambda, v_h) = -\int_K \Delta e_H v_h + \int_{\partial K\setminus\partial\Omega}\Big(\operatorname{sign}(s)\lambda - \frac{\partial u_H}{\partial \boldsymbol{n}_K}\Big)v_h + \\ + \int_{\partial K\cap\Gamma_N}\Big(g - \frac{\partial u_H}{\partial \boldsymbol{n}_K}\Big)v_h \qquad \forall\, v_h \in M_h(K)\,. \quad (3.4.8)$$

where we have used the fact that λ vanishes on $\partial\Omega$. In particular, by selecting $v_h = \varphi_K^\lambda$, we obtain the energy norm $\|\varphi_K^\lambda\|_{E,K}^2$. Since the union of the element contributions φ_K^λ realizes the infimum, by inserting such value into the Lagrangian, and recalling that $\sup(-\chi) = -\inf\chi$, we conclude that

$$\|P_H e_H\|_E^2 = -2\sup_{\lambda}\inf_{v_h\,\in\,\mathscr{M}_h} L(v_h,\lambda) = -2\sup_{\lambda} L(\varphi_K^\lambda,\lambda) = \\ = -2\sup_{\lambda}\Big(-\sum_K \tfrac{1}{2}\|\varphi_K^\lambda\|_{E,K}^2\Big) = \inf_{\lambda}\Big(\sum_K \|\varphi_K^\lambda\|_{E,K}^2\Big)\,.$$

The elemental problem (3.4.8) suggests to identify λ with the flux over the edge. Without any further insight, for each element K, the most reasonable choice is the average approximate flux:

$$\operatorname{sign}(s)\lambda = \tfrac{1}{2}\left(\frac{\partial u_H}{\partial \boldsymbol{n}_K} + \frac{\partial u_H^*}{\partial \boldsymbol{n}_K}\right) =: \left\langle \frac{\partial u_H}{\partial \boldsymbol{n}_K} \right\rangle ,$$

where u_H^* is the approximate solution in the adjacent elements. Recalling that $\Delta e_H = \Delta u - \Delta u_H = -(f + \Delta u_H) =: -r_h$, which is called element residual, the local problem becomes:

$$\begin{aligned}
&\text{Find } \varphi_K^\lambda \in M_h(K) \text{ such that:}\\
&B_K(\varphi_K^\lambda, v_h) = \int_K r_h v_h - \int_{\partial K \setminus \partial\Omega} \tfrac{1}{2} \left[\!\left[\frac{\partial u_H}{\partial \boldsymbol{n}_K} \right]\!\right] v_h + \int_{\partial K \cap \Gamma_N} \left(g - \frac{\partial u_H}{\partial \boldsymbol{n}_K} \right) v_h \\
&\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad \forall\, v_h \in M_h(K) \,,
\end{aligned}$$

where we recall that the jump has been defined as $[\![\partial u_H/\partial \boldsymbol{n}_K]\!] = \partial u_H/\partial \boldsymbol{n}_K - \partial u_H^*/\partial \boldsymbol{n}_K$.

Let's use the implicit error estimator we just discussed to draw a comparison to the explicit error estimator we previously developed. The second order derivatives present in the element residual can be eliminated by integrating the term $\int_K \Delta u_H v_h$ by parts:

$$B_K(\varphi_K^\lambda, v_h) = \int_K (f v_h - \nabla u_H \cdot \nabla v_h) + \int_{\partial K \setminus \partial\Omega} \tfrac{1}{2} \left\langle \frac{\partial u_H}{\partial \boldsymbol{n}_K} \right\rangle v_h + \int_{\partial K \cap \Gamma_N} g v_h \,.$$

While this circumvents the need for the second order derivatives of the geometry map, we are still left with a jump-like term. In the case of meshes where an (active) element face can have a number of smaller neighbors on the other side, the integration of jump-like terms is the bulk of the implementation. Furthermore, in the case of an implicit error estimator we need to develop shape functions for the space of bubble functions $M_h(\hat{K})$. This is a significant effort in terms of coding, hence another reason for choosing an explicit error estimator.

The reliability of an error estimator is ultimately assess through numerical experiments. In particular, we need to verify that the error estimator behaves as the actual error in terms of convergence rates. Numerical experiments concerning the error estimator are the subject of the next section.

3.5 Element Routine Testing

We verified the implementation of acoustics/elasticity element routine and the error estimator through three model problems: *(i)* free vibrations of an elastic sphere subjected to a Neumann load; *(ii)* free vibrations of an elastic hollow sphere filled with an acoustic fluid, and subjected to a Neumann load; *(iii)* acoustical wave propagation in a spherical annulus, to verify the PML.

3.5.1 Model Problems

I Problem. We assume the elastic displacement to be irrotational, i.e., $\boldsymbol{u} = \nabla\phi$, where ϕ is a scalar potential to be determined. The equation of elasto-dynamics yields:

$$\begin{aligned}\mathbf{0} &= -\varrho_s\omega^2\boldsymbol{u} - \operatorname{div}(E\,\nabla\boldsymbol{u}) = -\varrho_s\omega^2\boldsymbol{u} - [(\lambda+\mu)\,\nabla(\operatorname{div}\boldsymbol{u}) + \mu\,\Delta\boldsymbol{u}] = \\ &= -\varrho_s\omega^2\nabla\phi - [(\lambda+\mu)\,\nabla(\operatorname{div}\nabla\phi) + \mu\,\Delta\nabla\phi] = -\nabla\big[\varrho_s\omega^2\,\phi + (\lambda+2\mu)\,\Delta\phi\big]\,.\end{aligned}$$

In the previous chain of equality we used the vector identity $-\Delta\boldsymbol{\psi} = \operatorname{curl}\operatorname{curl}\boldsymbol{\psi} - \nabla(\operatorname{div}\boldsymbol{\psi})$, which in the case $\boldsymbol{\psi} = \nabla\phi$ reduces to $\Delta\nabla\phi = \nabla\Delta\phi$. A sufficient condition for the above equation to be satisfied is that the potential satisfies the Helmholtz equation:

$$\frac{\varrho_s\omega^2}{\lambda + 2\mu}\,\phi + \Delta\,\phi = 0\,.$$

We set $k_e^2 = \varrho_s\omega^2/(\lambda + 2\mu)$ and seek a potential of the form $\phi = f(r)/r$, where r is the radial coordinate and f is a function to be determined. Then f satisfies the equation $k_e^2 f + f'' = 0$, hence the elastic displacement is:

$$\begin{aligned}\boldsymbol{u} = \frac{d}{dr}\bigg(A\,\frac{\sin(k_e r)}{r} + B\,\frac{\cos(k_e r)}{r}\bigg)\boldsymbol{e}_r\ &= \\ = A\bigg(\frac{k_e r\ \cos(k_e r) - \sin(k_e r)}{r^3}\bigg)\boldsymbol{x} &+ B\bigg(\frac{-k_e r\ \sin(k_e r) - \cos(k_e r)}{r^3}\bigg)\boldsymbol{x}\,.\end{aligned}$$

Since the function associated to coefficient B yields a singular displacement at the origin, which is unphysical, it must be discarded, i.e., $B = 0$; we set $A = 1$. We drive the problem through a Neumann boundary condition; we choose the following non-dimesional material constants:

$$\omega = 1.1 \qquad ; \qquad \varrho_s = 1 \qquad ; \qquad \mu = 1 \qquad ; \qquad \lambda = 1.5\,.$$

We employ an initial mesh of eight tetrahedra only, one for each octant of the sphere, so that we can perform multiple levels of uniform refinements. We simulate the problem with isoparametric elements of second and third order. The results are reported in Table 3.4. They are in agreement with the theoretical rate of convergence and show agreement between the rate of converge of the error and the one of the error indicator θ_e. In a second test, in order to perform further levels of refinements and

# dof's	$\|\boldsymbol{u}-\boldsymbol{u}_h\|_1$	$\dfrac{\|\boldsymbol{u}-\boldsymbol{u}_h\|_1}{\|\boldsymbol{u}\|_1}$	θ_e	α_{err}	α_{est}
$p=2$					
75	0.21295 E−02	0.009156	0.80345 E−01		
441	0.12320 E−02	0.004873	0.35844 E−01	−0.308900	−0.455626
2865	0.38669 E−03	0.001519	0.10654 E−01	−0.619257	−0.648360
20529	0.10483 E−03	0.000412	0.28390 E−02	−0.662811	−0.671544
155505	0.26983 E−04	0.000106	0.72925 E−03	−0.670261	−0.671253
$p=3$					
189	0.74973 E−03	0.002965	0.38228 E−01		
1299	0.22257 E−03	0.000874	0.86477 E−02	−0.630044	−0.771058
9099	0.36911 E−04	0.000145	0.14221 E−02	−0.923030	−0.927343
67659	0.49599 E−05	0.000019	0.19305 E−03	−1.000393	−0.995341

Table 3.4: Numerical results for I Problem. Uniform refinements were used.

to test irregular meshes, we use the so-called greedy strategy: we refine all elements K such that:

$$\|\boldsymbol{u}-\boldsymbol{u}_h\|_K^2 \geq \varepsilon \max_K \|\boldsymbol{u}-\boldsymbol{u}_h\|_K^2 \,.$$

We choose as a threshold value $\varepsilon = 0.3$, and use second order elements. The results are reported in Table 3.5 and plotted in Figure 3.6(a). The actual error and the error indicator exhibit the same behavior and their rates of convergence approach the theoretical one as refinements are performed.

# dof's	$\|\boldsymbol{u}-\boldsymbol{u}_h\|_1$	θ_e	α_{err}	α_{est}	% ref. elem's
$p = 2$					
75	0.21295 E−02	0.80345 E−01			100.0
441	0.12320 E−02	0.35844 E−01	−0.308900	−0.455626	87.5
2499	0.42147 E−03	0.11429 E−01	−0.618401	−0.658934	15.8
3849	0.36621 E−03	0.93852 E−02	−0.325336	−0.456199	25.0
12111	0.18995 E−03	0.48317 E−02	−0.572684	−0.579201	7.3
18933	0.11336 E−03	0.29713 E−02	−1.155257	−1.088204	2.4
21201	0.10430 E−03	0.27709 E−02	−0.736246	−0.616998	39.0
72333	0.62005 E−04	0.15482 E−02	−0.423786	−0.474319	8.8
125517	0.34196 E−04	0.87678 E−03	−1.079695	−1.031616	

Table 3.5: Numerical results for I Problem. Adaptive refinements were used.

II Problem. The elastic displacement is obtained as in the previous problem:

$$\boldsymbol{u} = \frac{d}{dr}\left(A\,\frac{\exp(ik_e r)}{r} - B\,\frac{\exp(-ik_e r)}{r}\right)\boldsymbol{e}_r = \\ = A\,\frac{\exp(ik_e r)}{r}\left(ik_e - \frac{1}{r}\right)\boldsymbol{e}_r + B\,\frac{\exp(-ik_e r)}{r}\left(ik_e + \frac{1}{r}\right)\boldsymbol{e}_r \,.$$

Similarly, since the acoustic pressure p satisfies the Helmoltz equation $k_a^2\,p + \Delta\,p = 0$, with $k_a = \omega/c_0$, setting $p = f(r)/r$ yields:

$$p = C\,\frac{\sin(k_a r)}{r} + D\,\frac{\cos(k_a r)}{r} \,.$$

The function associated to constant D is singular at the origin, hence we set $D = 0$. We select a solution in the acoustic fluid, i.e., setting $C = 1$, and determine constants A and B of the elastic displacement by the interface conditions. In the case of a spherical geometry, the interface conditions reduce to:

$$
\begin{aligned}
&-p = \sigma_{rr} = 2\mu \frac{du_r}{dr} + \lambda\left(2\frac{u_r}{r} + \frac{du_r}{dr}\right) , \\
&\frac{dp}{dr} = \omega^2 \varrho_f u_r ,
\end{aligned}
$$

where $u_r = \boldsymbol{u} \cdot \boldsymbol{e}_r$. Hence we get the linear system

$$
\begin{aligned}
&A\,\frac{\exp(ik_e r)}{r}\left(ik_e - \frac{1}{r}\right) + B\,\frac{\exp(-ik_e r)}{r}\left(ik_e + \frac{1}{r}\right) = \frac{1}{\omega^2 \varrho_f}\frac{k_a r\cos(k_a r) - \sin(k_a r)}{r^2} , \\
&A\,\frac{\exp(ik_e r)}{r}\left[-(2\mu+\lambda)k_e^2 - 4\mu\left(\frac{ik_e}{r} - \frac{1}{r^2}\right)\right] + \\
&\qquad B\,\frac{\exp(-ik_e r)}{r}\left[(2\mu+\lambda)k_e^2 - 4\mu\left(\frac{ik_e}{r} + \frac{1}{r^2}\right)\right] = \frac{\sin(k_a r)}{r} .
\end{aligned}
$$

which needs to be satisfied at the interface, yielding appropriate values for A and B. As before, the problem is driven through a Neumann condition imposed on the outer boundary of the hollow sphere. We employ a mesh of eight tetrahedra, one per octant, for the acoustic sphere and two layers of prisms, for a total of sixteen prismatic elements, for the elastic hollow sphere, see Figure 3.7(a). We simulate the problem with isoparametric elements of second and third order, and the following non-dimensional material constants:

$$
\omega = 1.1 \quad ; \quad \varrho_s = 1 \quad ; \quad \mu = 1 \quad ; \quad \lambda = 1.5 \quad ; \quad \varrho_a = 1 \quad ; \quad c_0 = 1 .
$$

Let us recall that, when in the asymptotic region, we are only guaranteed an error estimate for the total error, namely:

$$
\begin{aligned}
\|(\boldsymbol{u},p) - (\boldsymbol{u}_h,p_h)\|_{1,\Omega_e\times\Omega_a} &= \left(\|\boldsymbol{u}-\boldsymbol{u}_h\|^2_{1,\Omega_e} + \|p-p_h\|^2_{1,\Omega_a}\right)^{1/2} \leq \\
&\leq C h^{\min\{p,r\}}\left(\|\boldsymbol{u}\|^2_{r+1,\Omega_e} + \|p\|^2_{r+1,\Omega_a}\right)^{1/2} .
\end{aligned}
$$

Thus, it only makes sense to compare the rate of convergence of the total error in elasticity and acoustics to that one of the total error indicator $\theta = (\theta_e^2 + \theta_a^2)^{1/2}$. Table 3.6 reports the results for the case of uniform refinements. The observed rates of convergence agree with the theoretical ones.

Things are more delicate in the case of adaptive refinements. In the case of a coupled problem, we deal with two different sets of unknowns (displacement and pressure, in our case) measured in different energy spaces. Even with a proper non-dimensionalization, the energy errors and norms corresponding to elastic or acoustic subdomains may have dramatically different values. As we will see, in the actual head problem contributions from the acoustical domain to the total (non-dimensional) energy are orders of magnitude larger than those coming from the elastic domain. Hence, the resolution of the elastic part of the domain becomes of secondary importance for any energy-based h-adaptive scheme.

As observed by Matuszyk et al. in [57], the key point for a successful application of an energy-based adaptive strategy to a coupled acoustics/elasticity problem is to rescale the governing equations to balance the acoustic and elastic energy norms of the solution. In the case of a weak coupling, this can be achieved through a proper choice of the preconditioning factor we discussed in Section 3.2. We resort to the simpler strategy of replacing each element error by the normalized error:

$$\|\boldsymbol{u} - \boldsymbol{u}_h\|_{1,K}^2 \leftarrow \frac{\|\boldsymbol{u} - \boldsymbol{u}_h\|_{1,K}^2}{\|\boldsymbol{u} - \boldsymbol{u}_h\|_{1,\Omega_e}^2} \qquad ; \qquad \|p - p_h\|_{1,K}^2 \leftarrow \frac{\|p - p_h\|_{1,K}^2}{\|p - p_h\|_{1,\Omega_a}^2} \, .$$

Table 3.7 shows the results obtained when element errors are not rescaled and adaptive refinements are performed. Notice that the elasticity error and the acoustic error are of the same order of magnitude, as the elasticity error indicator and the acoustic error indicator are as well. This could explain why things are working rather well, even without rescaling the error. Notice, however, that between the sixth and seventh refinement level, the error decreases, as predicted, while the error indicator increases. The issue has at least two obvious causes: elements' errors were not rescaled, and only 2.5% of the total number of elements were refined in the process. We ran the same numerical experiments employing normalized elements' errors for driving adaptivity. The results are reported in Table 3.8 and plotted in Figure 3.7(b). Such strategy produces less refinements. This is consistent with what is observed for the next verification problem. This strategy eventually fails as well, between the eighth and ninth refinement level. However, in this case things go wrong after 0.1% of the total number of elements is refined! This is and unrealistic percentage, hence the following strategy suggests itself: use normalized errors, employ the greedy strategy *and* refine at least a set percentage of elements.

III Problem. As seen in the previous problem, by seeking a pressure $p = f(r)/r$, the Helmholtz equation $k_a^2 p + \Delta p = 0$ immediately yields as a solution:

$$p = A \frac{\exp(ik_a r)}{r} + B \frac{\exp(-ik_a r)}{r} .$$

This is a superposition of an incoming wave, associated to coefficient A, and an outgoing wave, associated to coefficient B. The objective of the PML is to eliminate the incoming wave, i.e., $A = 0$; we set $B = 1$. We employed a domain composed of two spherical shells, the inner shell being the acoustic fluid and the outer shell

# dof's	$\|\boldsymbol{u}-\boldsymbol{u}_h\|_1$	$\frac{\|\boldsymbol{u}-\boldsymbol{u}_h\|_1}{\|\boldsymbol{u}\|_1}$	$\|p-p_h\|_1$	$\frac{\|p-p_h\|_1}{\|p\|_1}$	θ_e	θ_a	α_{err}	α_{est}
$p=2$								
295	0.15563 E+00	0.127642	0.11785 E+00	0.063559	0.85310 E+00	0.66877 E+00		
2145	0.31091 E−01	0.022632	0.38625 E−01	0.018613	0.22747 E+00	0.21961 E+00	−0.690771	−0.621034
15841	0.79009 E−02	0.005696	0.11462 E−01	0.005474	0.57287 E−01	0.78965 E−01	−0.635289	−0.588107
121521	0.20034 E−02	0.001443	0.31447 E−02	0.001501	0.13674 E−01	0.22786 E−01	−0.646577	−0.638276
$p=3$								
861	0.17901 E−01	0.013010	0.17770 E−01	0.008548	0.12615 E+00	0.11375 E+00		
6871	0.33447 E−02	0.002411	0.58759 E−02	0.002805	0.30493 E−01	0.63789 E−01	−0.633892	−0.422020
52551	0.41029 E−03	0.000296	0.79520 E−03	0.000380	0.32691 E−02	0.82052 E−02	−0.994038	−1.022397
409271	0.51325 E−04	0.000037	0.10326 E−03	0.000049	0.39376 E−03	0.10616 E−02	−0.998237	−1.000773

Table 3.6: Numerical results for II Problem. Uniform refinements were used.

# dof's	$\|\boldsymbol{u}-\boldsymbol{u}_h\|_1$	$\|p-p_h\|_1$	θ_e	θ_a	α_{err}	α_{est}	% ref. elem's
$p=2$							
295	0.15563 E+00	0.11785 E+00	0.85310 E+00	0.66877 E+00			100.0
2145	0.31091 E−01	0.38625 E−01	0.22747 E+00	0.21961 E+00	−0.690771	−0.621034	33.3
3233	0.31033 E−01	0.23416 E−01	0.22508 E+00	0.13074 E+00	−0.593005	−0.474139	21.3
15849	0.79006 E−02	0.11431 E−01	0.57282 E−01	0.12287 E+00	−0.647156	−0.410346	23.1
18353	0.78858 E−02	0.78936 E−02	0.56725 E−01	0.58974 E−01	−1.496150	−3.441848	17.5
65475	0.56046 E−02	0.33680 E−02	0.32992 E−01	0.38940 E−01	−0.420153	−0.371147	5.1
117353	0.22387 E−02	0.31424 E−02	0.15411 E−01	0.44161 E−01	−0.904040	−0.149511	2.5
121889	0.20034 E−02	0.31409 E−02	0.13682 E−01	0.48600 E−01	−0.924018	2.016244	20.4
156213	0.19998 E−02	0.15961 E−02	0.13301 E−01	0.14446 E−01	−1.514273	−3.806198	11.8
495459	0.13479 E−02	0.84081 E−03	0.78278 E−02	0.10151 E−01	−0.412899	−0.369487	

Table 3.7: Numerical results for II Problem. Adaptive refinements were used.

# dof's	$\|\boldsymbol{u}-\boldsymbol{u}_h\|_1$	$\|p-p_h\|_1$	θ_e	θ_a	α_{err}	α_{est}	% ref. elem's
$p=2$							
295	0.15563 E+00	0.11785 E+00	0.85310 E+00	0.66877 E+00			100.0
2145	0.31091 E−01	0.38625 E−01	0.22747 E+00	0.21961 E+00	−0.690771	−0.621034	54.2
7061	0.29567 E−01	0.16331 E−01	0.25071 E+00	0.12201 E+00	−0.322197	−0.105534	7.8
11681	0.16757 E−01	0.11565 E−01	0.12344 E+00	0.11542 E+00	−1.005547	−0.994730	3.9
15881	0.79006 E−02	0.11429 E−01	0.57291 E−01	0.13191 E+00	−1.244186	−0.525338	29.1
33955	0.74189 E−02	0.60835 E−02	0.42297 E−01	0.58566 E−01	−0.487280	−0.905977	2.6
36985	0.72929 E−02	0.34614 E−02	0.41930 E−01	0.38129 E−01	−2.020305	−2.839735	5.9
68001	0.53571 E−02	0.33244 E−02	0.31371 E−01	0.43370 E−01	−0.405867	−0.093789	8.1
119961	0.21732 E−02	0.31327 E−02	0.14941 E−01	0.47543 E−01	−0.886060	−0.125908	0.1
122301	0.20034 E−02	0.31326 E−02	0.13677 E−01	0.52496 E−01	−1.295811	4.392011	25.9
237899	0.18056 E−02	0.17743 E−02	0.10612 E−01	0.21049 E−01	−0.577873	−1.252699	

Table 3.8: Numerical results for II Problem. Adaptive refinements and rescaled error norms were used.

being the PML. The problem is driven through a Neumann condition on the inner boundary. We employ the following non-dimensional material constants:

$$\omega = 1.1 \quad ; \quad \varrho_a = 1 \quad ; \quad c_0 = 1 \, .$$

Each shell is meshed with eight prismatic elements. We performe numerical experiments using isoparametric elements. In the PML the order of approximation was raised by 2 in the radial direction, to accommodate for high gradients induced by the complex stretching, see Figure 3.8(b). The approximate solution p_h obtained for $p = 2$ on a uniformly refined mesh is shown in Figure 3.8(c). A common bound for positive and negative values was used, so that color green corresponds to 0; it can be visually seen how the solution dies out in the PML.

Table 3.9 shows the results obtained in the case of uniform refinements for $p = 2$ and $p = 3$. Since we are using variable order elements, we expect the rate of convergence to exceed the theoretical rate of convergence for second and third order elements. This is indeed confirmed by our numerical experiments. We use the greedy strategy to perform adaptive h-refinements. The results are reported in Table 3.10 and show good agreement between rate of convergence of the error and the rate of convergence of the error estimator. We run the same experiment rescaling the errors in the acoustic domain and in the PML domain. Consistently with what we observed in the previous problem, this strategy produces a smaller number of refinements. The numerical results, see Table 3.11 and Figure 3.8(a), are analogous to those obtained in the case of non-rescaled element errors.

# dof's	$\|p - p_h\|_1$	$\frac{\|p - p_h\|_1}{\|p\|_1}$	θ	α_{err}	α_{est}
$p = 2$					
126	0.13609 E+02	0.805328	0.92580 E+02		
966	0.19624 E+01	0.232785	0.77465 E+01	−0.950740	−1.217956
7326	0.50748 E+00	0.062351	0.21282 E+01	−0.667537	−0.637688
57006	0.92140 E−01	0.011357	0.34712 E+00	−0.831572	−0.883828
450126	0.22123 E−01	0.002727	0.88444 E−01	−0.690444	−0.661680
$p = 3$					
342	0.74183 E+01	0.643994	0.10251 E+03		
2806	0.89326 E+00	0.111290	0.64904 E+01	−1.005759	−1.311159
21790	0.17319 E+00	0.021353	0.56680 E+00	−0.800364	−1.189485
171086	0.92899 E−02	0.001145	0.28334 E−01	−1.419628	−1.453840
1355822	0.43375 E−03	0.000053	0.19947 E−02	−1.480296	−1.281920

Table 3.9: Numerical results for III Problem. Uniform refinements were used.

# dof's	$\|p - p_h\|_1$	θ_a	α_{err}	α_{est}	% ref. elem's
$p = 2$					
126	0.13609 E+02	0.92580 E+02			100.0
966	0.19624 E+01	0.77465 E+01	−0.950740	−1.217956	31.2
3204	0.99926 E+00	0.73732 E+01	−0.562884	−0.041190	5.9
5058	0.55382 E+00	0.38337 E+01	−1.292642	−1.432508	31.9
13932	0.26921 E+00	0.29011 E+01	−0.711934	−0.275075	6.4
21358	0.12652 E+00	0.15978 E+01	−1.767334	−1.396105	0.6
22650	0.11465 E+00	0.14500 E+01	−1.678170	−1.652789	47.7
86720	0.54561 E−01	0.48018 E+00	−0.553091	−0.823188	6.0
139050	0.35274 E−01	0.31774 E+00	−0.923788	−0.874567	2.4
165176	0.29076 E−01	0.28329 E+00	−1.122214	−0.666482	

Table 3.10: Numerical results for III Problem. Adaptive refinements were used.

# dof's	$\|p-p_h\|_1$	θ_a	α_{err}	α_{est}	% ref. elem's
$p=2$					
126	0.13609 E+02	0.92580 E+02			100.0
966	0.19624 E+01	0.77465 E+01	-0.950740	-1.217956	81.2
5256	0.97969 E+00	0.71232 E+01	-0.410089	-0.049519	2.8
7326	0.50748 E+00	0.30265 E+01	-1.980875	-2.577732	25.8
17250	0.25017 E+00	0.27703 E+01	-0.825939	-0.103272	5.0
25402	0.12081 E+00	0.15915 E+01	-1.880902	-1.432203	0.4
26694	0.10831 E+00	0.14119 E+01	-2.201757	-2.412841	14.8
48180	0.85478 E−01	0.12206 E+01	-0.400877	-0.246578	0.6
54812	0.79501 E−01	0.73039 E+00	-0.562056	-3.981980	9.8
107302	0.44338 E−01	0.42285 E+00	-0.869297	-0.813668	

Table 3.11: Numerical results for III Problem. Adaptive refinements and rescaled error norms were used.

3.5.2 Air Force problem

The actual problem is to be solved for multiple right-hand-sides. Specifically, we want to hit the head model with 13 incident plane pressure waves having directions corresponding to $\pi/6$ increments in the (x,y)-plane, starting from $(1,0,0)$, i.e., at normal incidence on the tympanic membrane or, equivalently, parallel to the ear canal. In the case of multiple right-hand sides the h-adaptive strategy needs to be revisited. The idea is to group together incident directions, and assume the behavior of the solution to be similar in nature within each group, so that a common refinement strategy can be applied. It is natural to take as an indicator the sum of the squares of the indicators for each direction. Table 3.12 reports the error indicator for each incident direction. We group together direction 1 through 4, 5 through 8, and 9 through 13.

direction	$\|p-p_h\|_1^2$	$\|\boldsymbol{u}-\boldsymbol{u}_h\|_1^2$	$\|p-p_h\|_{\mathrm{PML}}^2$	θ^2
1	0.17383 E−03	0.12354 E−12	0.14261 E−02	0.75909 E+07
2	0.17383 E−03	0.12302 E−12	0.14260 E−02	0.77646 E+07
3	0.17383 E−03	0.12206 E−12	0.14265 E−02	0.78791 E+07
4	0.17384 E−03	0.12092 E−12	0.14276 E−02	0.79370 E+07
5	0.17385 E−03	0.11990 E−12	0.14289 E−02	0.79783 E+07
6	0.17385 E−03	0.11928 E−12	0.14301 E−02	0.80665 E+07
7	0.17386 E−03	0.11923 E−12	0.14308 E−02	0.82653 E+07
8	0.17386 E−03	0.11975 E−12	0.14309 E−02	0.86160 E+07
9	0.17385 E−03	0.12070 E−12	0.14304 E−02	0.91198 E+07
10	0.17384 E−03	0.12184 E−12	0.14293 E−02	0.97317 E+07
11	0.17384 E−03	0.12285 E−12	0.14280 E−02	0.10367 E+08
12	0.17383 E−03	0.12347 E−12	0.14268 E−02	0.10918 E+08
13	0.17382 E−03	0.12353 E−12	0.14261 E−02	0.11279 E+08

Table 3.12: Distribution of error indicator over incident directions.

We employed the error indicator to perform several levels of refinements and study convergence rates of the error indicator. We used the material data from Table 3.1, with the following choice for the skull:

$$\text{skull:} \qquad E = 6500\,\mathrm{MPa} \quad , \quad \nu = 0.22 \quad , \quad \varrho = 1412\,\mathrm{kg/m}^3 \; .$$

The frequency was set to $f = 2000\,\mathrm{Hz}$. We solved for the first group of incident directions (1–4), and used a greedy strategy with $\varepsilon = 0.25$ combined with the refinement of at least 0.2 % of the elements. The data are plotted in Figure 3.9. We performed 25 levels of adaptive refinements reaching roughly 2.5 million degrees of freedom. The bulk of the refinements were in the PML, the skull, the air, and the brain. During the whole precess refinements were performed in 10 sub-damains out of a total of 16 sub-domains. This constitutes a challenging test for the refinement package and the error indicator. After a pre-asymptotic region, the error indicator approaches the theoretical rate of convergence.

Since we are interested in the pressure exerted over and interface rather than at a point, it is natural to define the (average) pressure on a surface S as:

$$p = \fint_S \Re\big[P(x)\, e^{i\omega t}\big] = \fint_S \big(\Re P(x)\cos(\omega t) - \Im P(x)\sin(\omega t)\big) =$$
$$= \left[\left(\fint_S \Re P\right)^2 + \left(\fint_S \Im P\right)^2\right]^{1/2} \cos(\omega t + \phi)\ .$$

The last equality is an immediate consequence of setting $A = \fint_S \Re P$, $B = \fint_S \Im P$, and $\cos\phi = A/(A^2+B^2)^{1/2}$, $\sin\phi = B/(A^2+B^2)^{1/2}$.

Although our first numerical experiment showed the correct behavior of the error indicator as h-refinements are performed, in practice the values of the scattered acoustic pressure over interfaces of interest of the model, e.g., the tympanic membrane, the oval and round window, and the basilar membrane, do not change between successive levels of refinements. In other words, we can capture the value of the average pressure on the interfaces of interest with the initial mesh. While this seems disappointing at first, it can only be realized afterwards; furthermore, it can be explained by the fact that we are employing high order elements in the initial mesh. On the implementation side, this is a great simplification. Since refinements are not needed, we can solve for all incident directions simultaneously. The most notable consequence is that investigation of the whole range of audible frequencies becomes feasible.

3.6 Parametric Study

The objective is to study the mechanics of the ear over the whole range of audible frequencies, which is roughly 20 Hz through 20 kHz. Frequencies below 500 Hz are referred to as *low* frequencies, frequencies between 500 Hz and 6 kHz are referred to as *mid* frequencies, and frequencies above 6 kHz are referred to as *high* frequencies. More specifically, we want to investigate the behavior of the pressure exerted on the basilar membrane as material constants, frequency, and direction of the incident pressure wave vary. Although we could also report the normal elastic displacement of the basilar membrane, we will in general favor the pressure since this is the quantity that is reported in most experimental studies. In fact, the pressure in the cochlea can be measured by *just* inserting a probe into the inner ear of a willing subject!

We computed the average acoustic pressure on the following interfaces for different scenarios:

(i) inner/outer tympanic membrane;

(ii) inner/outer round window;

(iii) inner/outer oval window;

(iv) upper/lower base, mid, and apex of basilar membrane (cf. Figure 3.11).

In order not to overwhelm the reader with data, we only report the result that we believe are interesting and relevant to the discussion. Beside pressure distributions, which are the main focus, we also computed the elastic displacement on the basilar membrane. The basilar membrane has variable stiffness and different frequencies are supposed to excite different parts of the membrane. Frequencies at the high end of the audible spectrum stimulate the base, while frequencies at the low end of the spectrum stimulate the apex, see Figure 3.11. We set the Young's modulus following the approach of [36]: the value decreases linearly from 50 MPa to 15 MPa between the base and the middle, and from 15 MPa to 5 MPa between the middle and the apex, see Figure 3.11. If the Young's modulus is changing within an element, the tensor of elasticity needs to be recomputed at each integration point. Although this would require only a minor modification of the element routine, the computations would become significantly more expensive. In fact, for each element we determine the Young's modulus by computing its value at the centroid of the element.

An equal-loudness contour, see Figure 3.10, is a measure of the Sound Pressure Level (SPL) over the frequency spectrum, for which a listener perceives a constant loudness when presented with pure steady tones. The SPL, which is measured in Decibel [dB], is defined as

$$\mathrm{SPL} = 20 \log_{10}(p_{\mathrm{rms}}/p_{\mathrm{ref}}) \, .$$

The standard value for the reference pressure p_{ref} is $20 \cdot 10^{-6}$ Pa when in the air, and 10^{-6} Pa when in the water. The pressure root-mean-square p_{rms} is defined as the average over the period of p^2. The unit of measurement for loudness levels is the *phon*, and is arrived at by reference to equal-loudness contours. By definition two sine waves of different frequencies are said to have equal loudness level, measured in phons, if they are perceived as equally loud by the average young person without significant hearing impairment. We see that the human auditory system is most sensitive to frequencies between 2 kHz and 5 kHz; this is largely due to the resonance of the ear canal and the transfer function of the ossicles of the middle ear.

As a first experiment, we pick $E = 6500\,\mathrm{MPa}$, $\nu = 0.22$, and $\varrho = 1412\,\mathrm{kg/m^3}$ as material constants for the skull and sweep over 201 frequencies, equally spaced on a logarithmic scale between 20 Hz and 20 kHz. The results for the pressure distributions over the basilar membrane when at normal incidence are reported in Figure 3.12. We observe that through the low and mid frequencies range the pressure exerted on the lower and upper interfaces of the basilar membrane does not change and is equal to roughly 81.5 Pa. When we reach about 10 kHz, we clearly see a resonant frequency of the system. As the frequency keeps increasing, the base, mid and apex of the basilar membrane start to exhibit different behaviors. A second resonant frequency is observed around 14 kHz. We concluded that the range of frequencies to which the ear model is sensitive is shifted towards the high frequencies.

If we decrease the Young's modulus for the skull to $E = 1600\,\mathrm{MPa}$ (cf. Table 3.1), see Figure 3.13, the model becomes more sensitive to low frequencies. The results are qualitatively similar to the previous case. We still observe two spikes in the pressure distribution over the basilar membrane. In order to better investigate the resonant frequencies of the system, we computed the normal elastic displacement at both ends of the ossicles, the one connected to the the tympanic membrane and the one connected to the oval window. The normal displacement is equal at both ends, thus the ossicles are acting as a rigid body. The normal elastic displacement is reported in Figure 3.14. The first resonant frequency of the system is around 5 kHz. At this frequency we also observe a spike in the normal displacement of the ossicles. The second resonant frequency is around 14 kHz, which is the same as in the previous case. At this frequency we do *not* see a spike in the normal displacement of the ossicles. The oval window is a stiff membrane, while the round window is a soft membrane that acts as a pressure releaser. We do observe this behavior. Figure 3.15 reports the difference in pressure between the inner and outer interface of the tympanic membrane, and the oval and round window in the case of normal incidence. Indeed the drop in pressure across the round window, i.e., the red curve, is close to zero. Low frequencies are supposed to excite the apex of the basilar membrane while high frequencies are supposed to excite the base. While we do not quite observe this behavior (cf. Figure 3.13), we do see a different pressure distribution on the apex, middle an base at frequencies higher than 10 kHz.

The previous numerical results suggest that by a careful manipulation of the material constants we could tweak our ear model to match–or, at least, resemble–the response of the actual ear. Since we are considering an idealized model, we believe that results that are qualitatively similar to the ones reported in the medical literature are already satisfactory. In the remainder of this section we investigate the effects of directionality, as well as blocking the ear canal with an ear plug. We limit ourself the frequencies between 3 kHz and 20 kHz.

We pick 5 frequencies between 3 kHz to 20 kHz and equally spaced on a logarithmic scale. Since on each graph we plot curves for all frequencies, we tailor the number of frequencies so that readability of the graphs is not compromised. We begin our discussion from the results obtained for the tympanic membrane. In Figure 3.16 we report the absolute value of the reflection coefficient $R := p^{\mathrm{sca}}/p^{\mathrm{inc}}$ and the power transmission coefficient $\tau := 1 - |R|^2$ for the outer interface. The plots show that the incident pressure wave is able to transmit most of its energy–more that 91%, the worst case scenario occurring at $f = 20\,\mathrm{kHz}$–to the tympanic membrane. In Figure 3.17 we report the pressure on the outer and inner interface. Recalling the discussion at the beginning of the chapter, the significant quantity on the outer interface is the total pressure (incident pressure plus acoustic, or scattered, pressure), while the significant quantity on the inner interface is the acoustic pressure, that can be interpreted as the transmitted pressure. At low frequency, when the wave length is much bigger than the aperture of ear canal, the ear canal acts like a side branch and is insensitive to the directionality of the incident plane wave. Indeed, when $f = 3\,\mathrm{kHz}$, the pressure at the bottom of the canal, i.e., the outer interface of the tympanic membrane, varies from 100.04 Pa at normal incidence, to 99.96 Pa at grazing incidence. As the frequency increases the dependence of the pressure upon directionality increases as well. The results for the pressure on the basilar membrane are illustrated in Figure 3.18. As observed for the tympanic membrane, the effect of directionality becomes more pronounced when the frequency increases. Furthermore, at the two highest frequencies, i.e., $f = 12447\,\mathrm{Hz}$ and $f = 20000\,\mathrm{Hz}$, besides the effect of directionality, we see a consistent difference in pressure between the upper and lower interface of the basilar membrane. As anticipated, the round window acts as a pressure releaser. While the previous experiment confirmed such behavior throughout the whole frequency spectrum, we can now investigate its dependence upon directionality. By comparing the two graphs on the bottom of Figure 3.19, which shows the pressure on both sides of the oval and round window, we do see that the round widow acts as a pressure releaser regardless of the incident direction. The two graphs in the left column show that the air in the middle ear is exerting roughly the same pressure on both the oval and round window. Contrary to the round window, the oval window is able to create a difference in pressure between its two sides, as seen by comparing the two graphs on the top. In the case of normal incidence, this difference is exactly the red curve plotted in Figure 3.15.

As a second experiment we investigate the effects of blocking the ear canal with a plug. For this purpose we employ the material constants of the E-A-R Classic Foam Earplug[1], which can be found in [45]:

$$\text{ear plug:} \qquad E = 22155\,\mathrm{Pa} \quad , \quad \nu = 0.1 \quad , \quad \varrho = 228\,\mathrm{kg/m}^3 \; .$$

All other material constants and details of the model are unchanged. In order to gain an insight into the response of the system, we swept over 201 frequencies, equally spaced on a logarithmic scale between 20 Hz and 20 kHz. The pressure distributions on the basilar membrane are qualitatively similar to those obtained in the case of the open ear canal, but they are shifted down, see Figure 3.20. In fact, as for the previous case, we observe two resonant frequencies. The lower one occurs at around 5 kHz, where the normal displacement of the ossicles exhibits a spike as well (cf. Figure 3.21), while the higher one occurs around 13 kHz. As in the previous case, we do not observe a distinctly different behavior of the base, middle and apex of the basilar

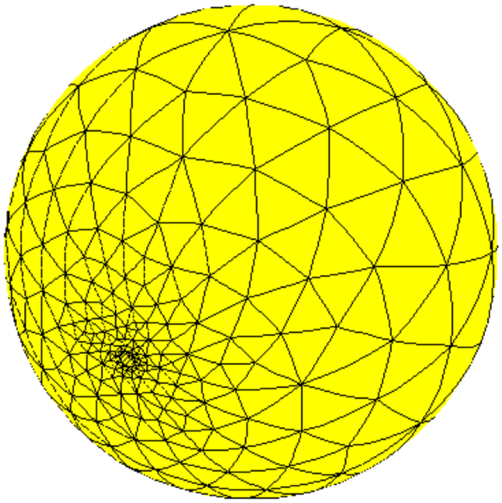
---

[1] this is a trademark of Aearo Corporation

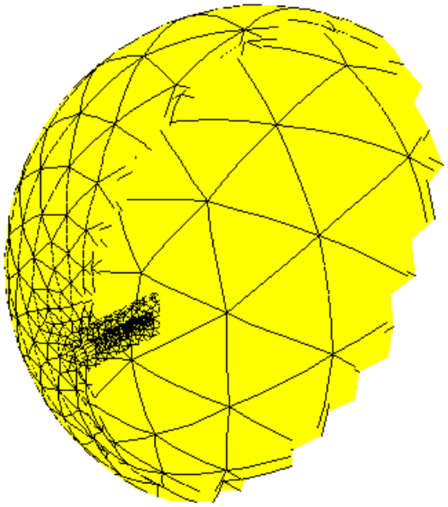
membrane to different frequencies. The difference in pressure between the inner and outer interface of the tympanic membrane, and the oval and round window in the case of normal incidence is reported in Figure 3.22. The round window is indeed acting as a pressure releaser, i.e., the red curve is close to zero.

As before, in order to investigate directionality, we pick 5 frequencies, ranging from 3 kHz to 20 kHz and equally spaced on a logarithmic scale. In Figure 3.23, we report the pressure on both sides of the plug. The pressure drops from roughly 100 Pa outside to 6 Pa inside the ear canal. The pressure on the basilar membrane is reported in Figure 3.24. As in the case of the open ear canal, we see that the effect of directionality increases as the frequency increases. In particular, when $f = 4821$, we are close to the lowest resonant frequency, hence the effect of directionality is magnified.

Given the flexible framework we have developed, many more numerical experiments could be conducted to investigate different scenarios. Validation of our numerical results against medical data would require an expertise that we are lacking and is beyond the scope of this work. Moreover, before embarking into elaborate simulations that could be meaningful under a clinical point of view, a close collaboration with people familiar with the physiology of the ear needs to be established. I hope to be able to follow this path and continue this research work in the future.

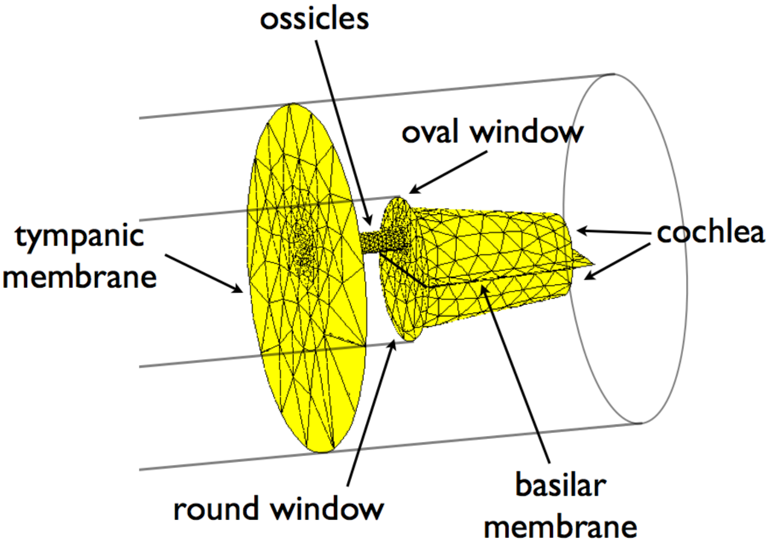


(a) Head model.

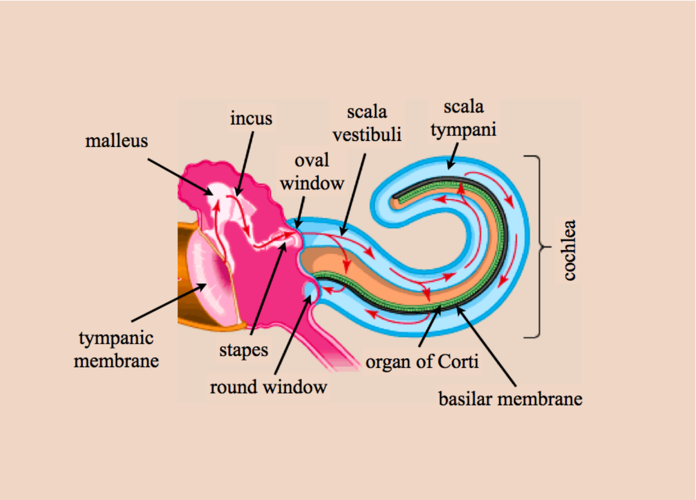


(b) Model cross-section.

Figure 3.1: Constructive Solid Geometry model built *via* Netgen and the Geometry Modeling Package.

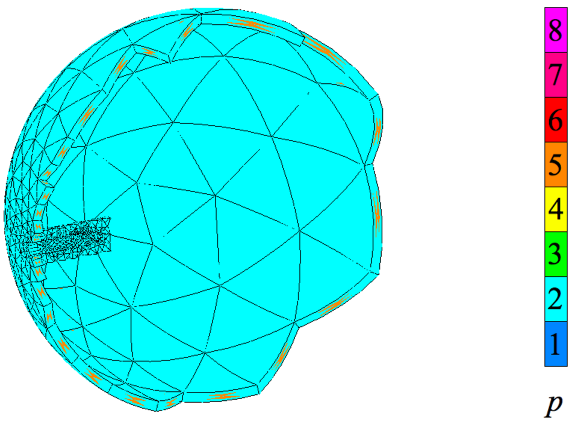


(a) Middle ear model.

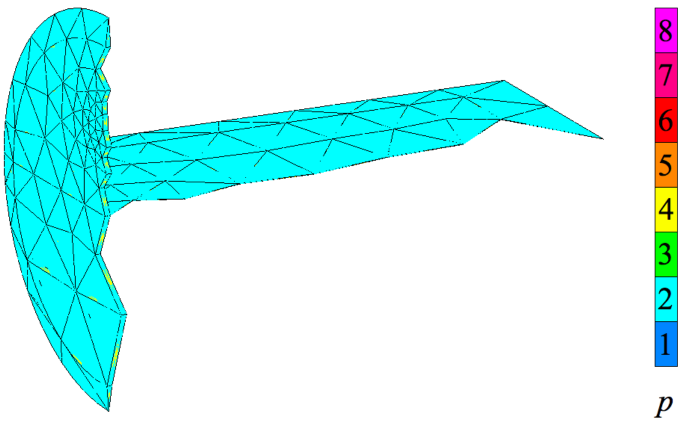


(b) Anatomy of the middle ear (image taken from the web).

Figure 3.2: Middle ear detail of the model.

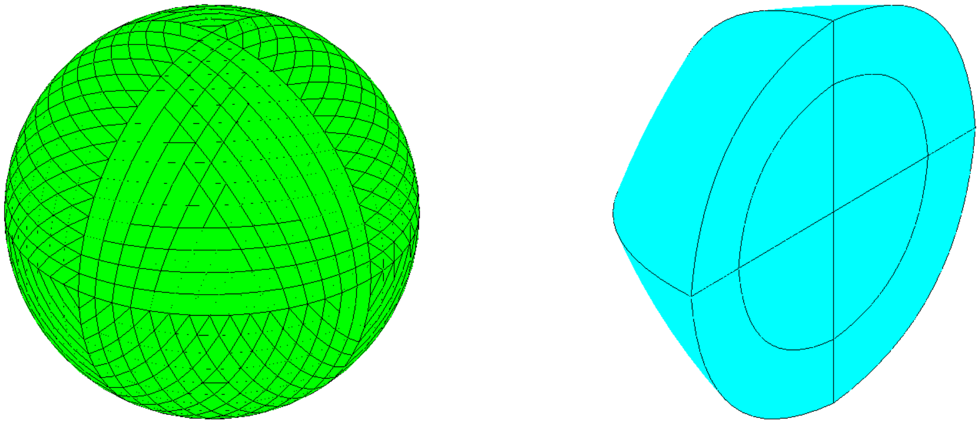


(a) Order of approximation is raised to $p = 5$ in for shell-like elements of the skull.



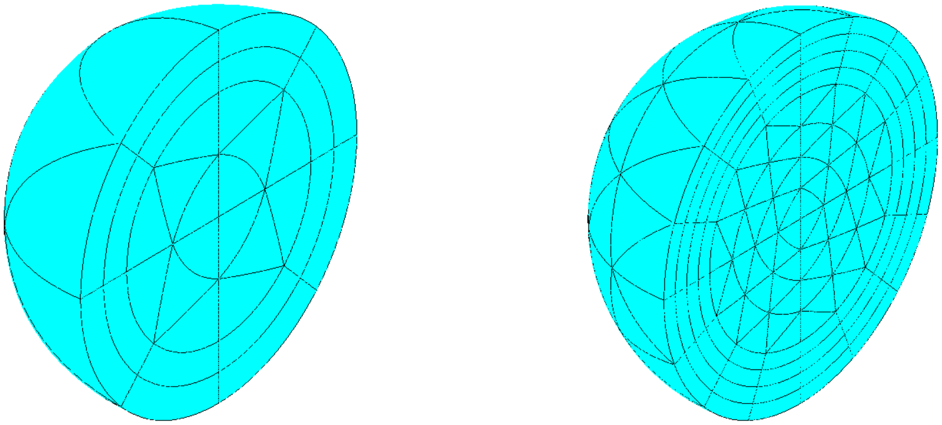
(b) Order of approximation is raised to $p = 4$ in the membranes.

Figure 3.3: Initial mesh order of approximation.



(a) Parameterization of the exact geometry map.

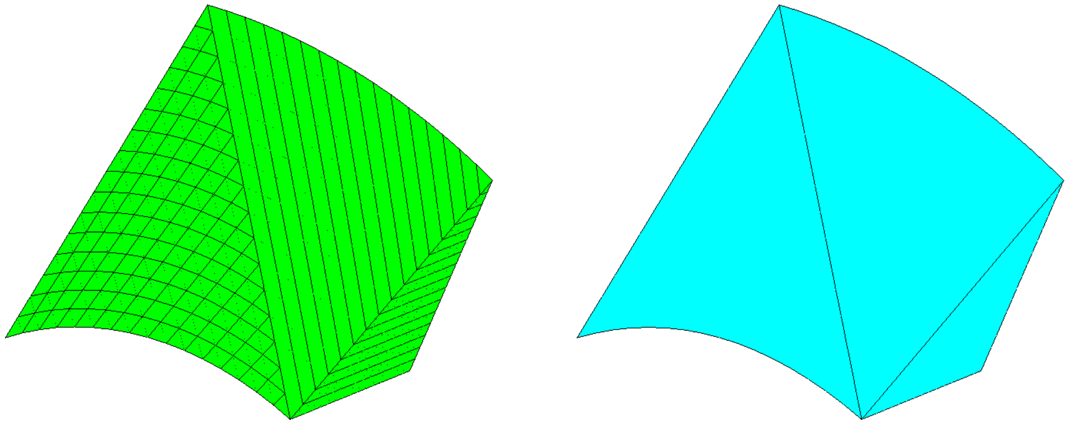
(b) Initial mesh cross-section.



(c) 1^{st} level of uniform refinements.

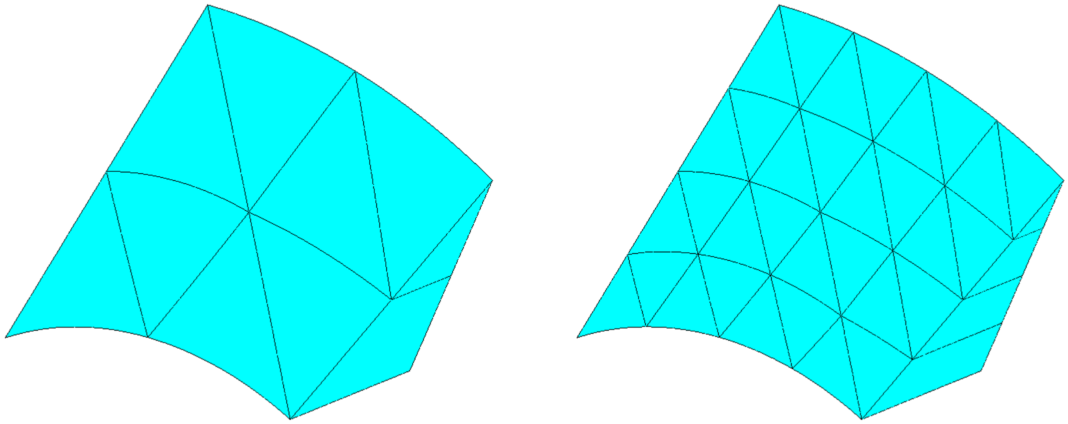
(d) 2^{nd} level of uniform refinements.

Figure 3.4: Refinements of a spherical mesh composed of eight tetrahedra and eight prisms.



(a) Parameterization of the exact geometry map.

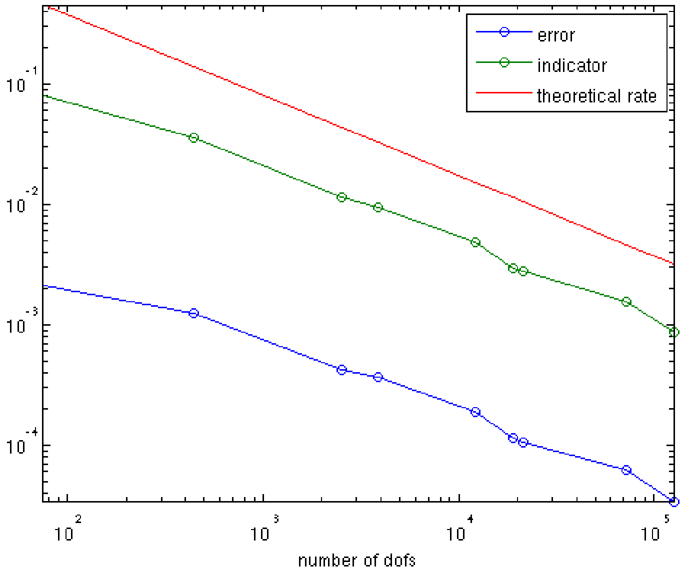
(b) Initial mesh.



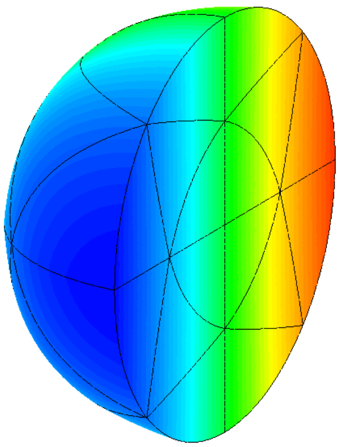
(c) 1st level of uniform refinements.

(d) 2nd level of uniform refinements.

Figure 3.5: Refinements of a portion of a circular membrane ribbon.

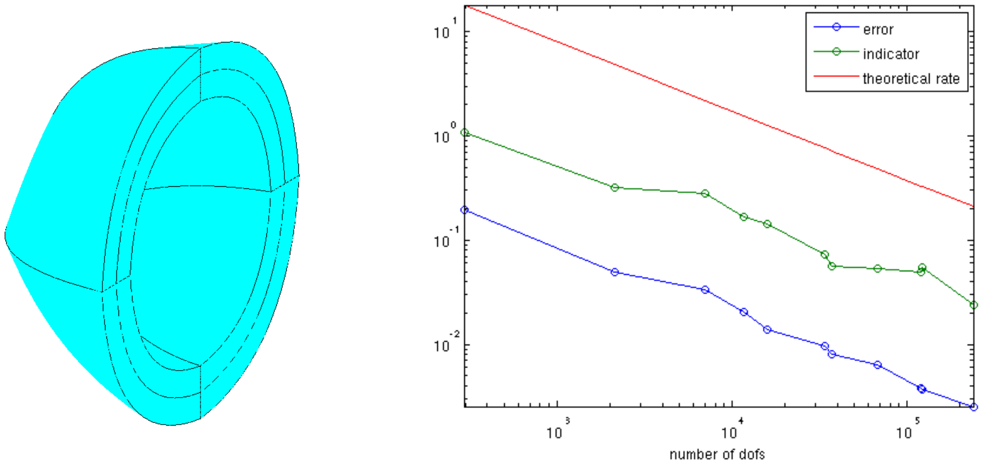


(a) Convergence rates.

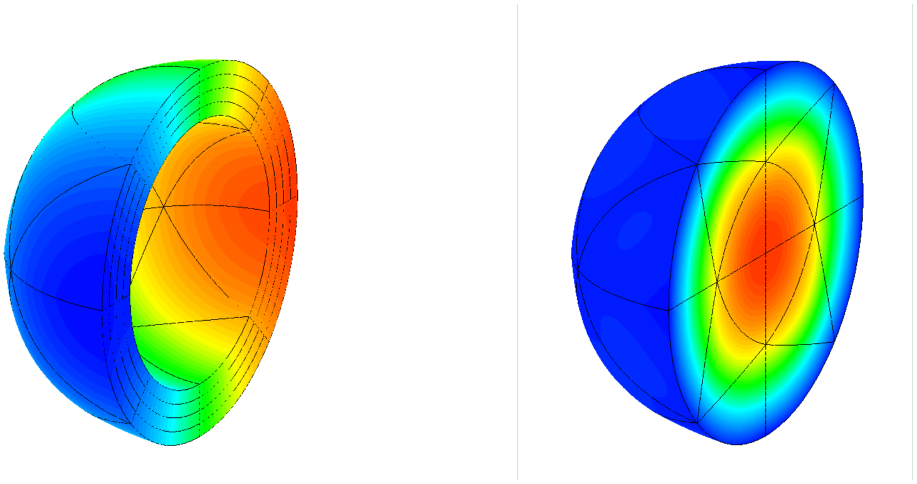


(b) u_1.

Figure 3.6: I Problem: elastic sphere subjected to normal load.

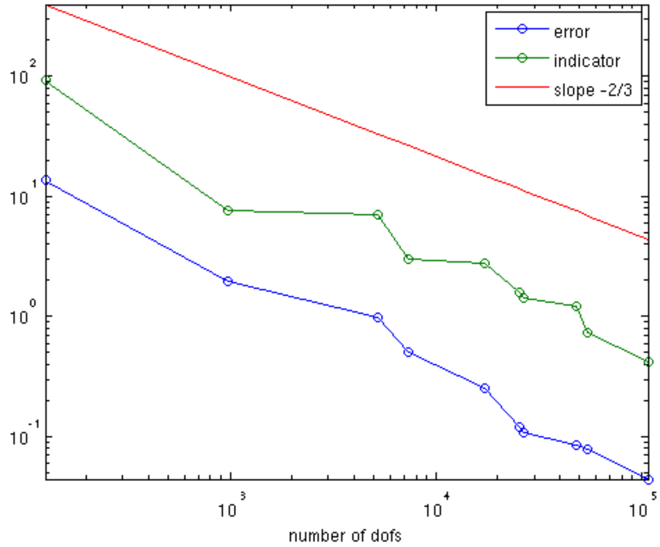


(a) Initial mesh cross-section.
(b) Convergence rates.

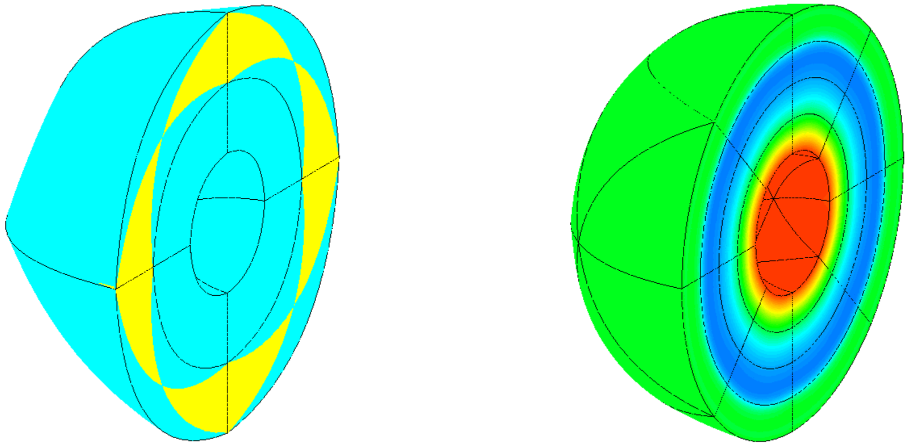


(c) u_1
(d) p^{sca}

Figure 3.7: II Problem: hollow elastic sphere filled with acoustic fluid.



(a) Convergence rates.



(b) Initial mesh cross-section.

(c) p^{sca}.

Figure 3.8: III Problem: acoustic fluid surrounded by PML.

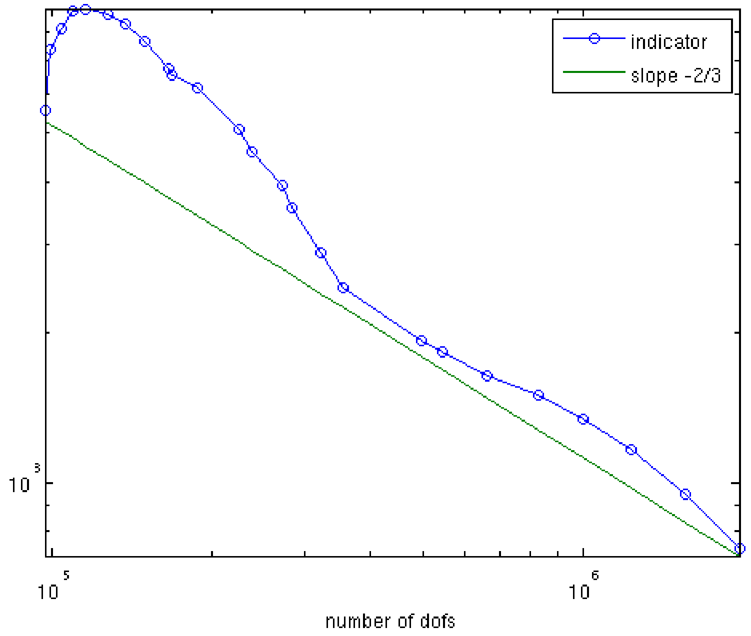


Figure 3.9: Error indicator convergence study.

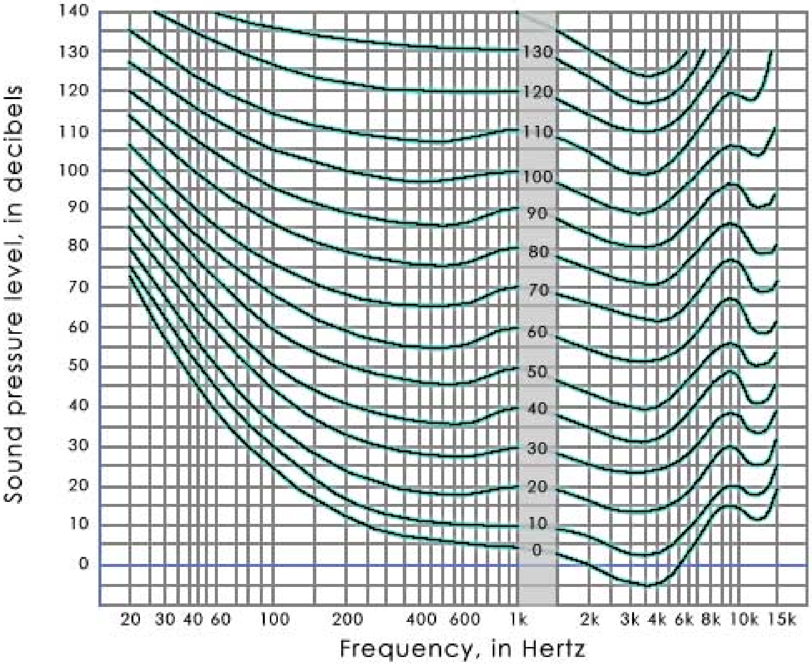


Figure 3.10: Equal loudness contours. Image taken from the web.

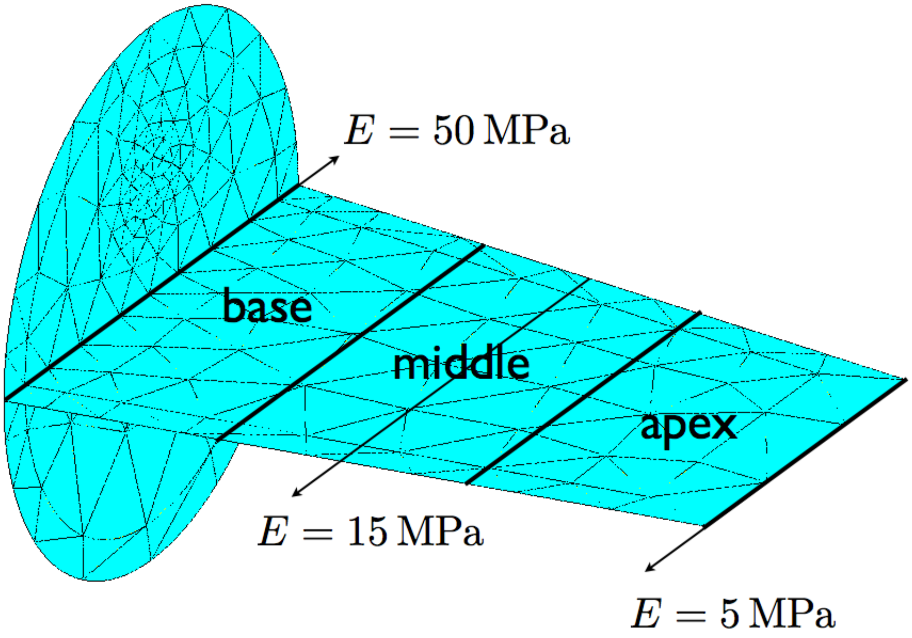


Figure 3.11: Base (adjacent to oval and round window), middle, and apex (at the far end of the cochlea) of the basilar membrane. The basilar membrane has variable Young modulus E.

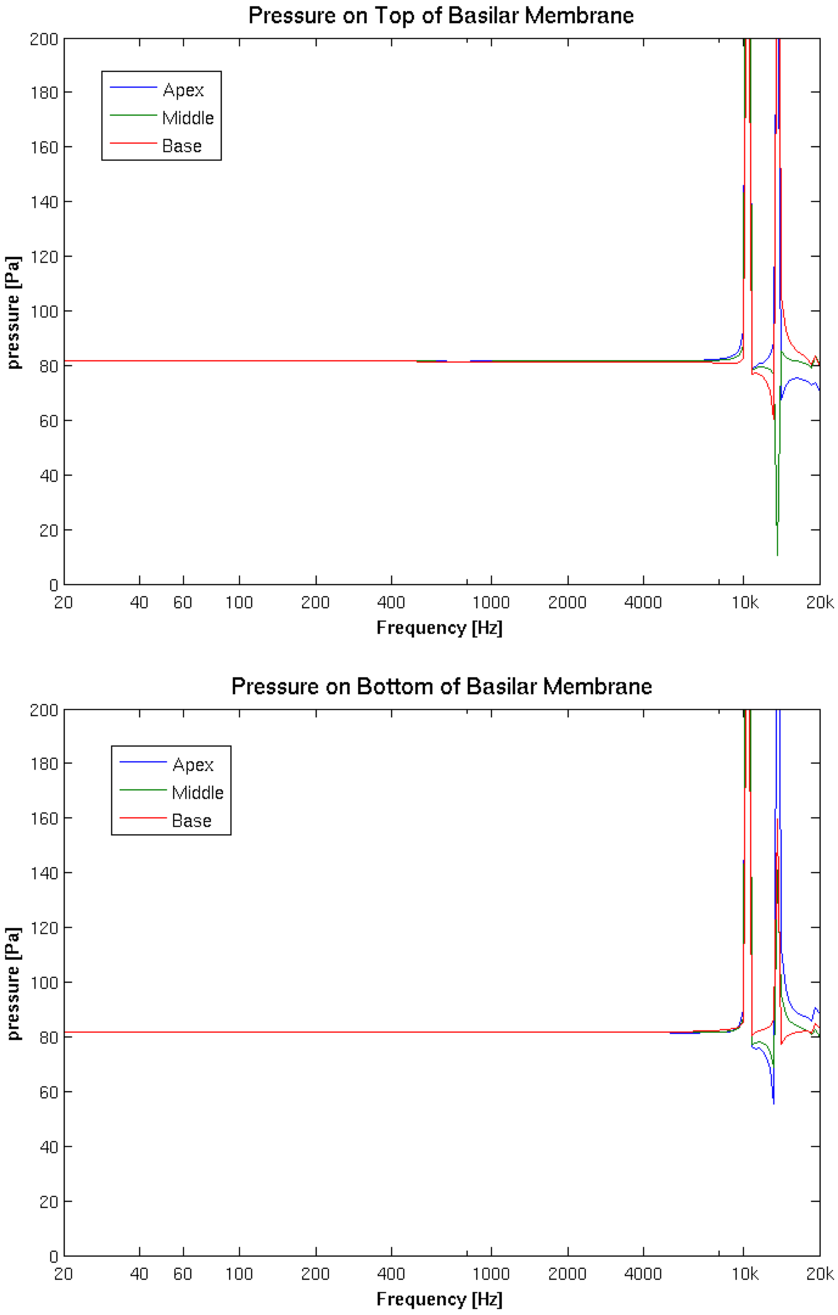


Figure 3.12: Pressure distributions over basilar membrane at normal incidence. Material constants $E = 6500\,\mathrm{MPa}$, $\nu = 0.22$, $\varrho = 1412\,\mathrm{kg/m^3}$ were used for the skull.

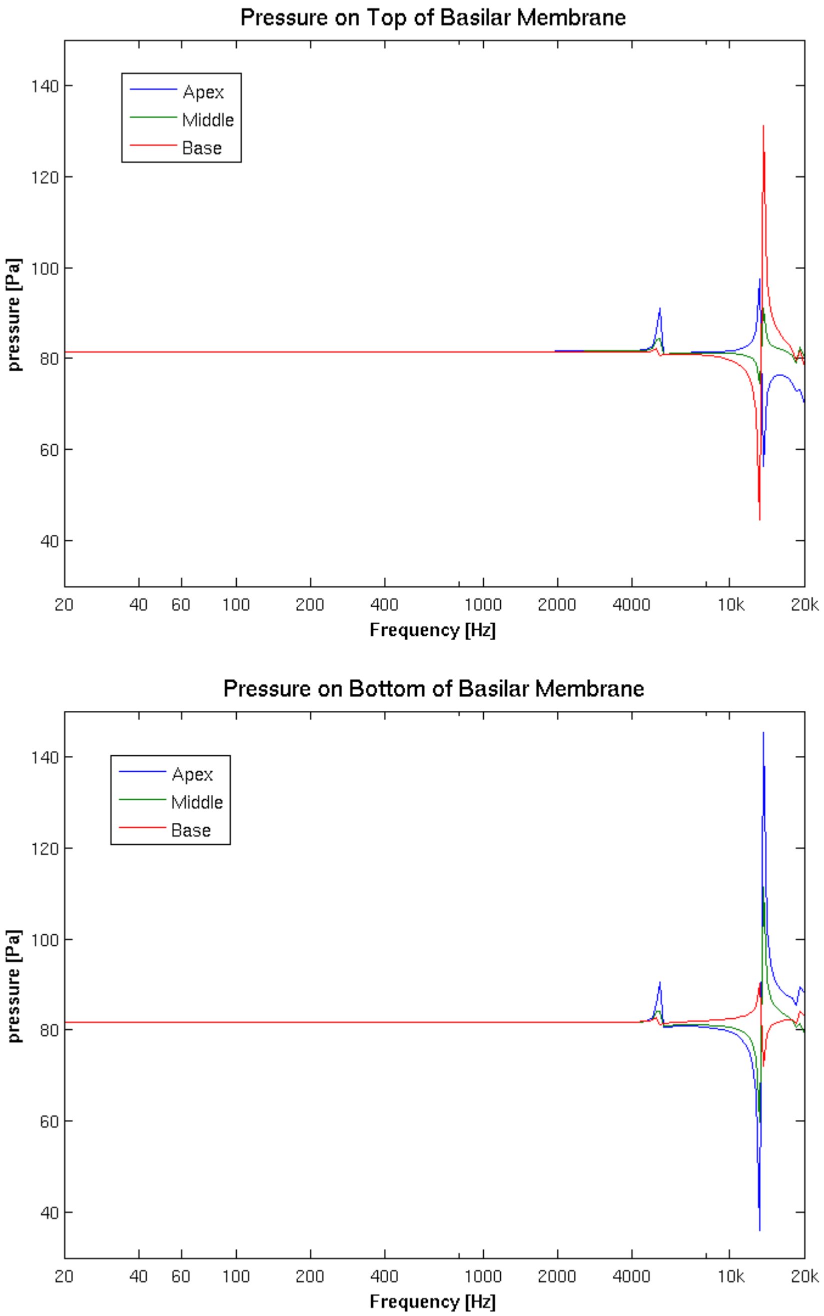


Figure 3.13: Pressure distributions over basilar membrane at normal incidence. Material constants $E = 1600\,\mathrm{MPa}$, $\nu = 0.22$, $\varrho = 1412\,\mathrm{kg/m^3}$ were used for the skull.

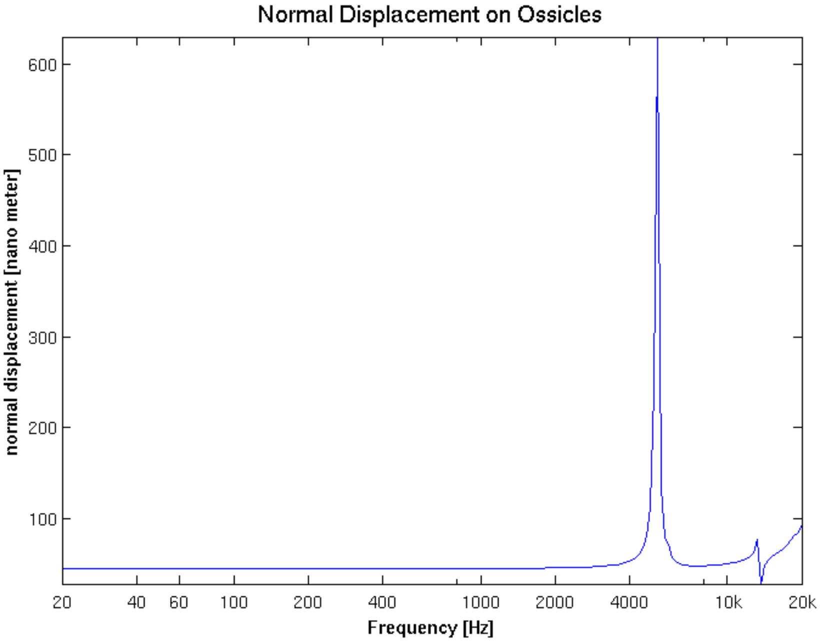


Figure 3.14: Ossicles normal displacement.

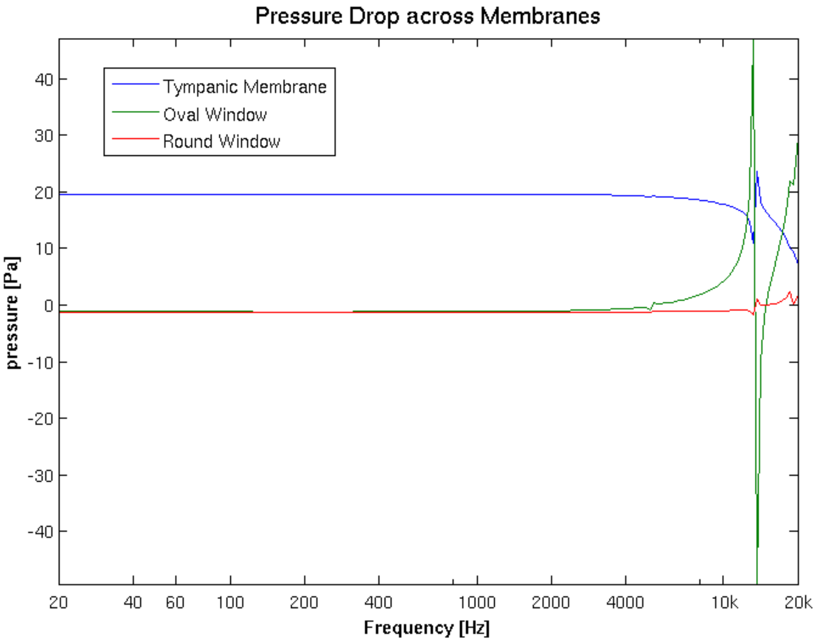


Figure 3.15: Change in pressure $(p_{\mathrm{out}} - p_{\mathrm{in}})$ across tympanic membrane, oval window and round window.

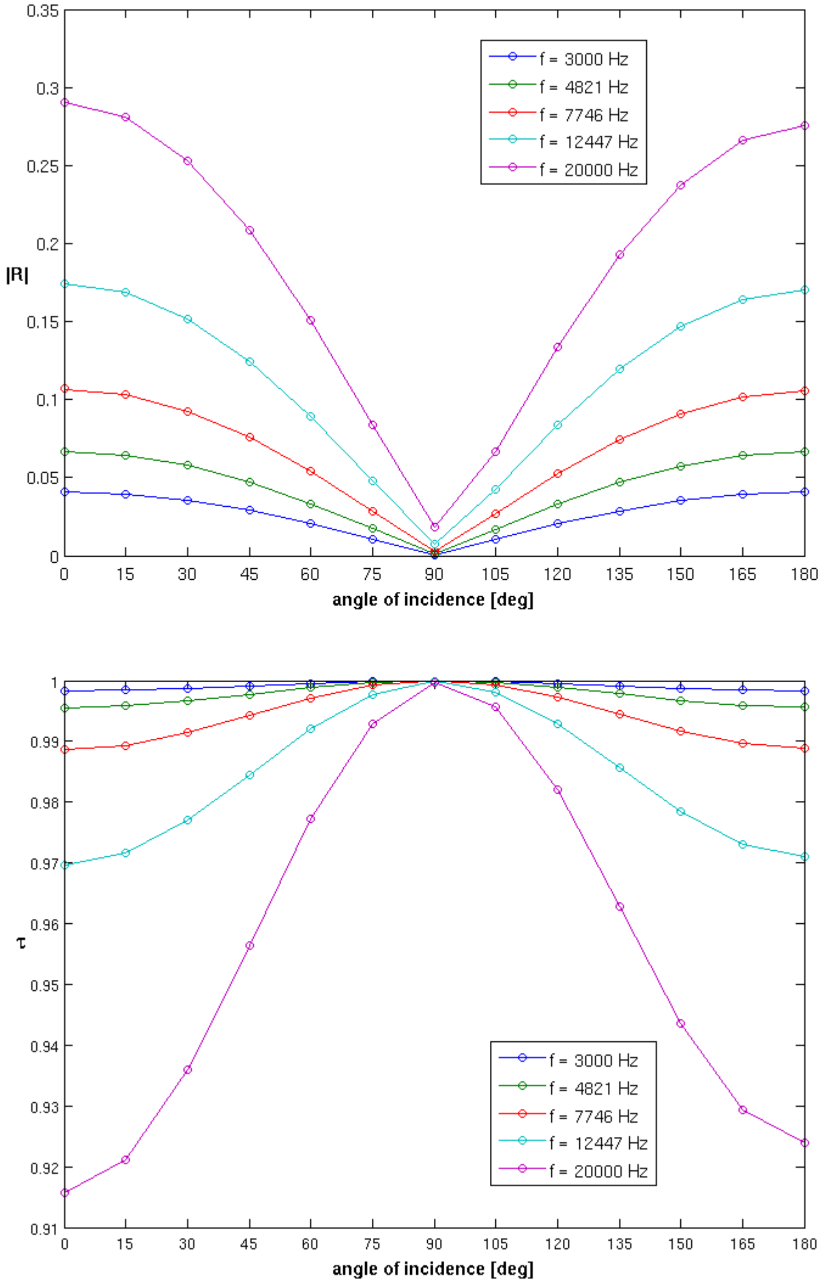


Figure 3.16: Absolute value of reflection coefficient R and power transmission coefficient τ on the outer interface of the tympanic membrane.

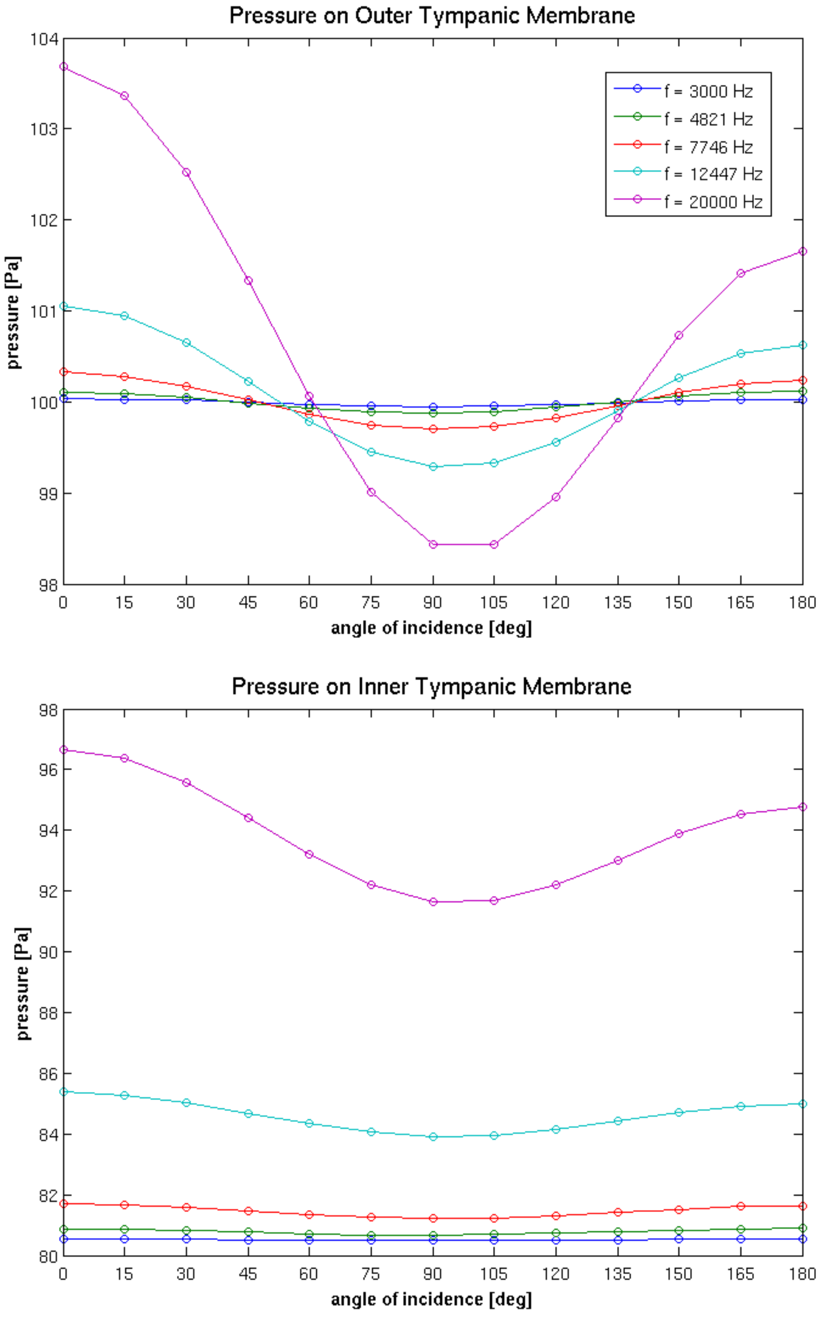


Figure 3.17: Pressure on the outer (above) and inner (below) interface of the tympanic membrane.

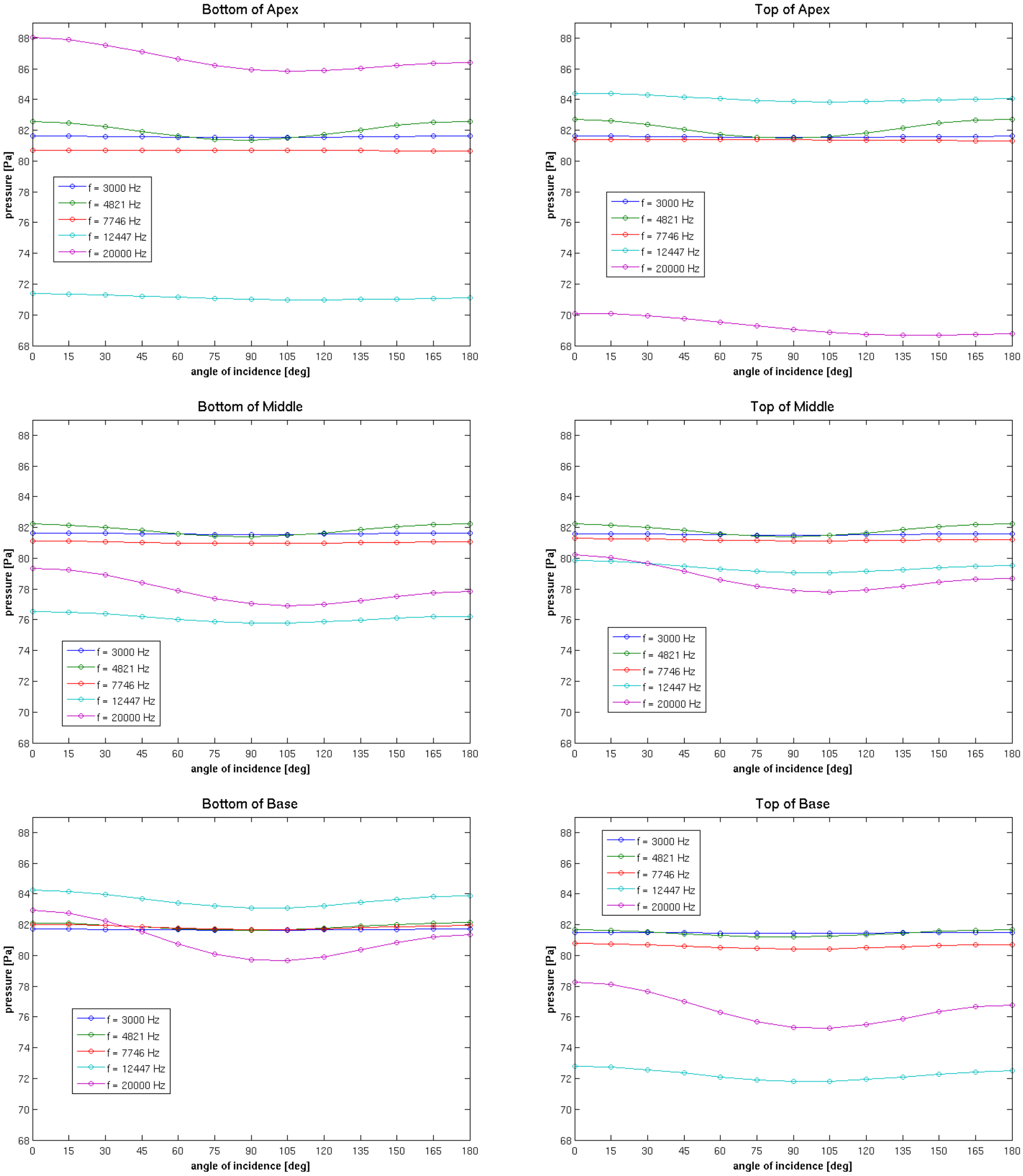


Figure 3.18: Pressure distribution on base, middle and apex of basilar membrane, cf. Figure 3.11.

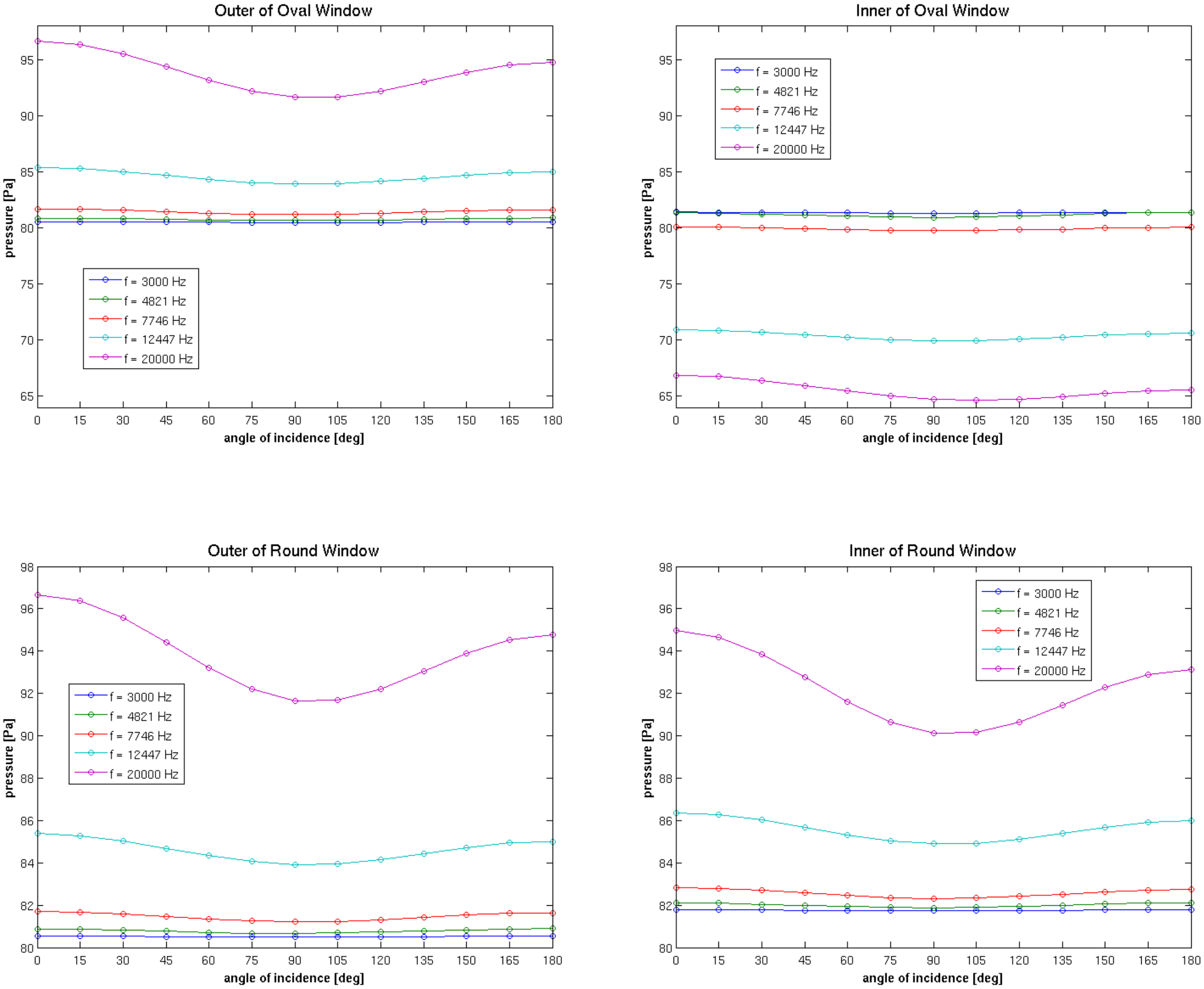


Figure 3.19: Pressure distribution oval window and round window.

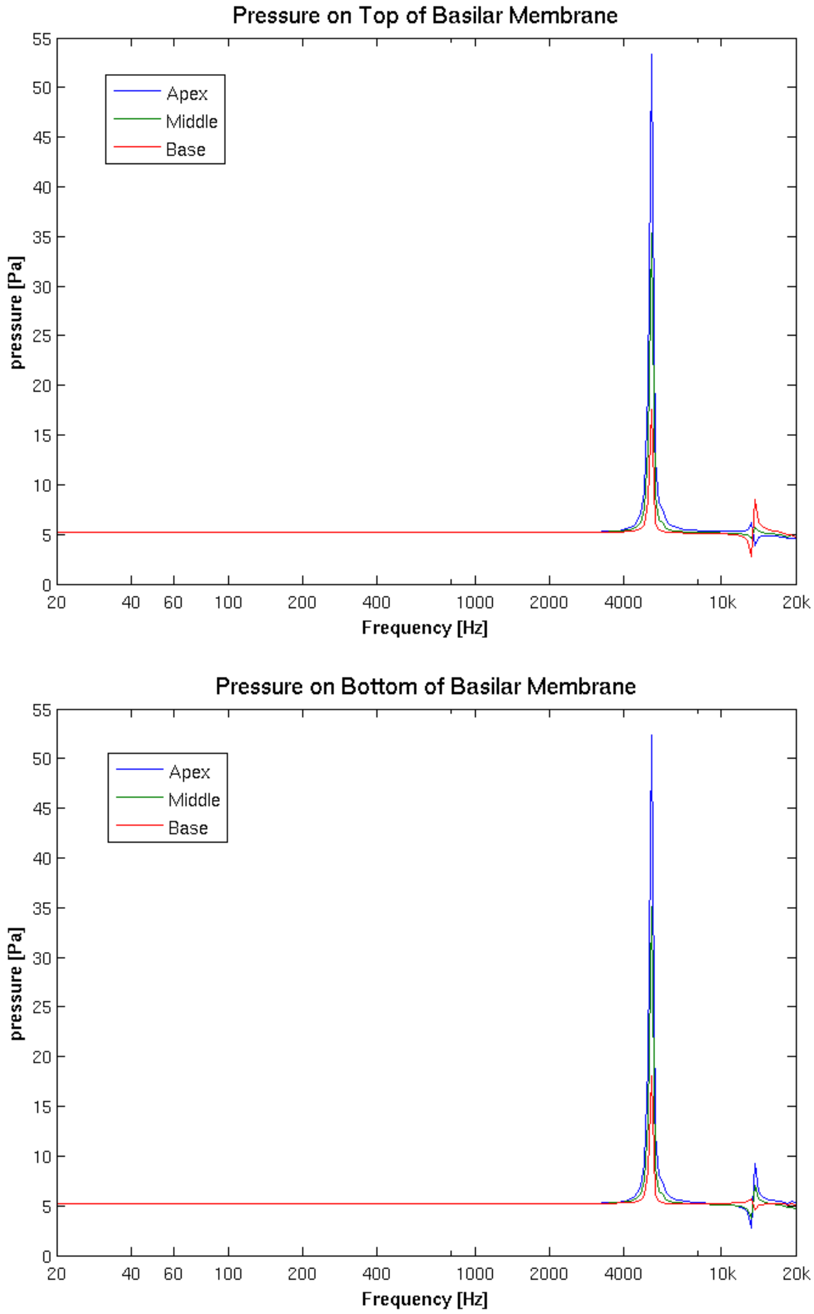


Figure 3.20: Pressure distributions over basilar membrane at normal incidence. Ear canal was blocked.

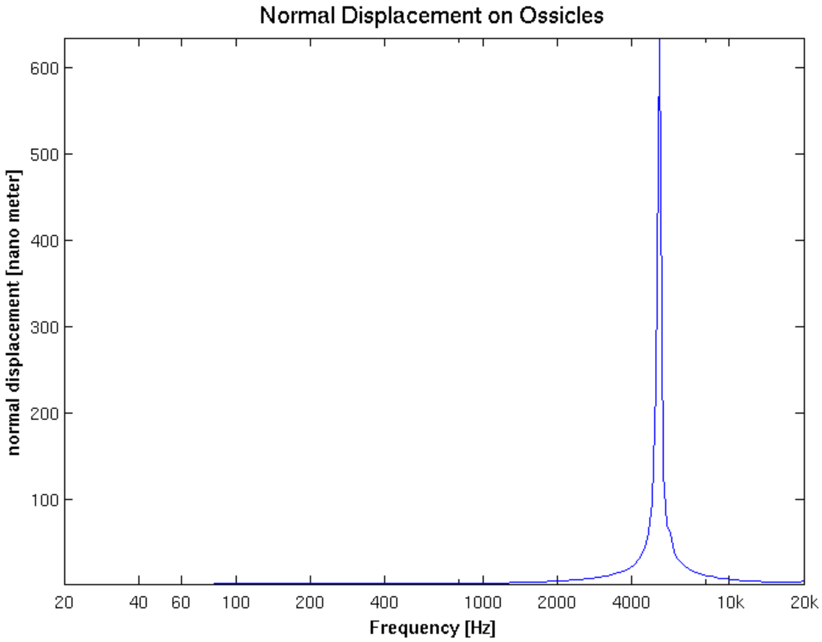


Figure 3.21: Ossicles normal displacement. Ear canal was blocked.

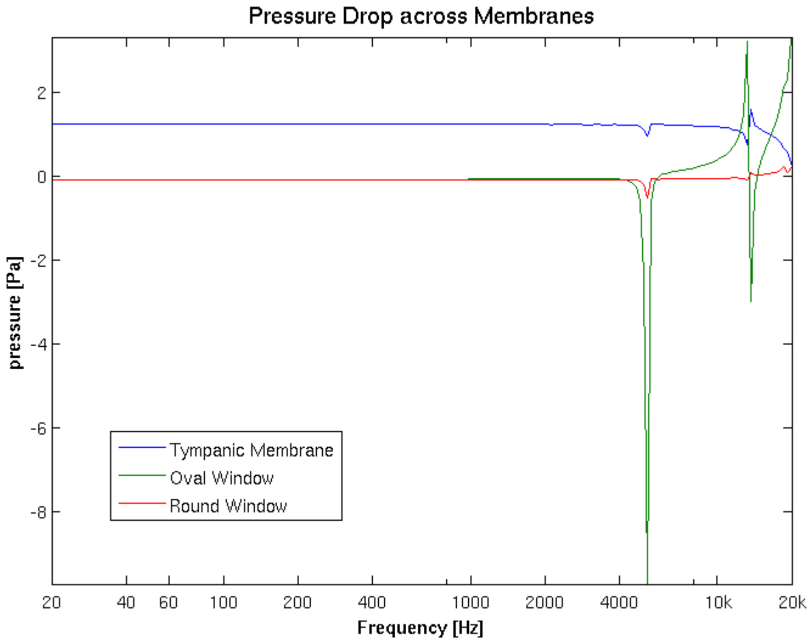


Figure 3.22: Change in pressure ($p_{\text{out}} - p_{\text{in}}$) across tympanic membrane, oval window and round window. Ear canal was blocked.

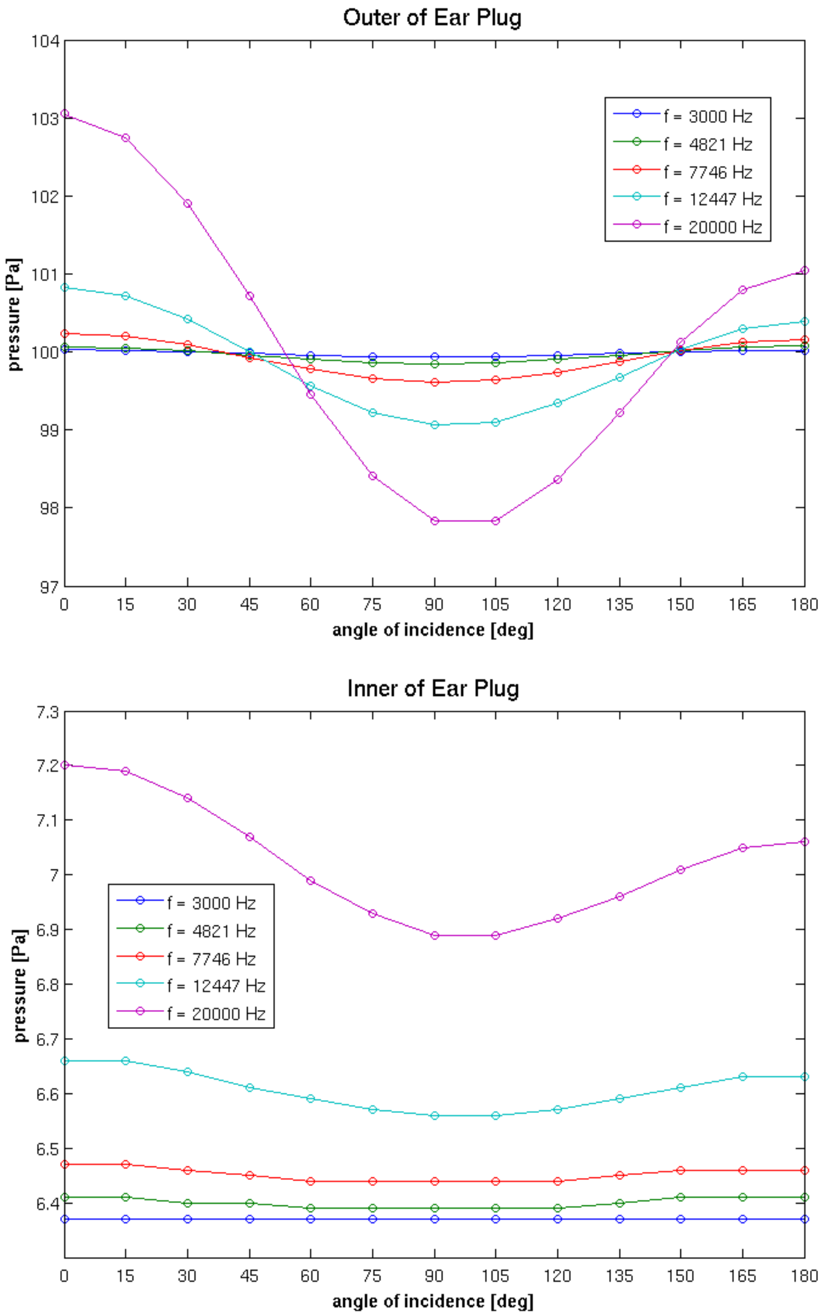
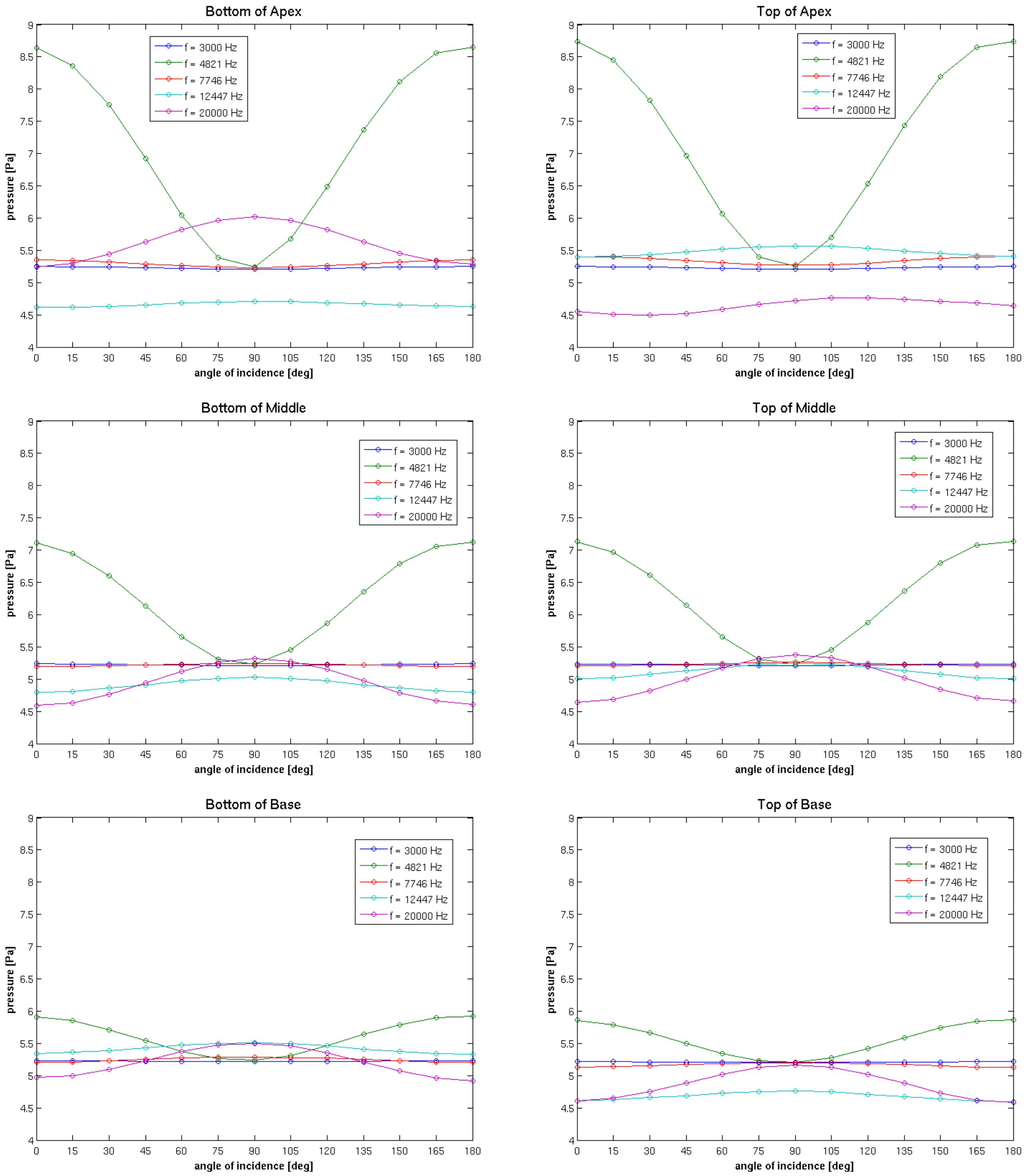


Figure 3.23: Pressure on the outer (above) and inner (below) interface of the ear plug.



Bottom of Apex

Top of Apex

Bottom of Middle

Top of Middle

Bottom of Base

Top of Base

Figure 3.24: Pressure distribution on base, middle and apex of basilar membrane (cf. Figure 3.11). Ear canal was blocked.

Chapter 4

Conclusions and Future Work

In the present work we considered a biomechanical application of a coupled acoustics/elasticity problem. The project presented us with a number of challenges, the most relevant of all being the software development. Using the experience accumulated over the past few decades by Demkowicz's research group in the field of *hp*-finite elements code implementation, we were able to build a flexible finite element library suitable for studying multi-physics problems.

Since we anticipated having to solve problems with millions of unknowns, we interfaced our FE library with the Multi-frontal Massively Parallel Sparse Direct Solver (MUMPS), a state-of-the-art solver that has gained vast popularity over the past few years. In this way we were able to start crunching numbers since the early stages of the project. Yet, the use of an external solver manifests a number of limitations arising from the fact that all insight into the structure of the application is lost. This fact prompted the development the Unassembled Hyper-Matrix Solver (UHM) by Kyungjoo Kim, a friend of mine and a fellow graduate student at The University of Texas at Austin. UHM is tightly integrated with the data structure of our FE library, specifically with the nodal trees that describe the mesh refinements. This enables UHM to exploit the opportunity for a factorization strategy that proceeds by updating the factorization from the previous *hp*-step, rather then having to recompute the whole factorization from scratch.

Although the development of the FE library was motivated by the Air Force project, we intend to use it as *the* code for tackling 3D problems with an industrial-level complexity. Indeed, the `hp3d` library is already being used by Kyungjoo to study the interaction of the human body, more specifically the head, with a nearby wireless device, i.e., a cellular phone. The problem requires to couple together the Maxwell's electromagnetic and Pennes' biological heat transfer equations.

This project has required a major redesign of the Geometry Modeling Package (GMP). I extended a previous version of GMP by adding support for a number of algebraic surfaces and blocks of four different shapes: tetrahedra, prisms, hexahedra, and the much neglected pyramids. Pyramids are necessary to connect a thin-walled structure, meshed with prisms, to a supporting structure meshed with tetrahedra. In practice, I built and implemented geometry parameterizations that map a reference block with straight edges and flat faces onto a physical block whose edges and faces must conform, respectively, to algebraic curves and surfaces. This is achieved through an incremental strategy. Edge and face bubbles are added to a linear or bilinear interpolant of the physical block in order to modify one edge at a time and, subsequently, one face at a time. Appropriate edge and face bubbles are constructed by Transfinite Interpolation and Parametric Transfinite Interpolation, namely Transfinite Interpolation in the space of parameters. For a detailed discussion see Section 2.1 and Gatto and Demkowicz [40].

In most of bioengineering applications, geometry is not known *a priori*, but it has to be reconstructed from various type of scans, e.g., Computer Tomography (CT) or Magnetic Resonance Imaging (MRI) scans. Typical geometry reconstruction techniques identify material interfaces and represent them in terms of unstructured triangular or quadrilateral grids which provide a starting point for meshing the volumes located in between the interfaces. The resulting tetrahedral or hexahedral grids provide only a very rough, piecewise linear or bilinear representation of the geometry, insufficient for higher order elements. Hence, it becomes necessary to upgrade the piecewise (bi)linear interface surfaces to smoother, higher-order reconstructions. This fact has motivated the development of reconstruction schemes for G^1-smooth (continuously changing normal) surfaces. While the case of reconstruction on quadrilateral grids is discussed in Demkowicz, Gatto, Qiu and Joplin [23], the case of triangular girds is ongoing research, see Appendix A.

Although we are confident that GMP has evolved into a mature and reliable software, further developments are certainly possible. A fairly easy task would be adding support for SPLINES and NURBS through interfaces with external libraries.

A Constructive Solid Geometry (CSG) object is obtained by union or intersection of a number of primitives, namely subdomains bounded by algebraic surfaces. We simulated the problem on a CSG object that resembles a real head model. A number of features of the middle ear model were designed to match the dimensions of the real middle ear that are found in medical literature. All membranes present in the model have thickness, so that inertial effects—which are believed to be crucial for the hearing mechanism—can be accounted for. This leads to a hybrid model which is composed of tetrahedra, prisms, and pyramids. Developing a code that accommodates elements of different shapes implies a number of challenges. First of all, for each shape, a hierarchical construction of shape functions needs to be developed, see Gatto and Demkowicz [40]. Those shape functions need to assembled into globally conforming basis functions, accounting for the different orientations of adjacent elements. In order to perform any sort of h-adaptivity, we need to be able to refine elements. We successfully implemented a refinement package that supports isotropic, as well as anisotropic, refinements for elements of four different shapes. This was by far the most complex refinement package ever implemented by Demkowicz's research group. We are not aware of any other code that can handle such a level of complexity as `hp3d`.

The accomplishments of the dissertation can by summarized as follow:

- A completely redesigned Geometry Modeling Package. More specifically, support for a number of algebraic surfaces and blocks of all shapes (tetrahedra, prisms, hexahedra and pyramids) was implemented from scratch. This requires to develop suitable parameterizations through Transfinite Interpolation and Parametric Transfinite Interpolation for a block edges and faces, see [40]. Furthermore, reconstruction schemes for G^1-smooth surfaces have been developed and implemented. We addressed both the case of quadrilateral, see [23], and triangular grids, see Appendix A.
- Construction of shape functions. A hierarchical construction of shape functions for the tetrahedron, hexahedron, prism and pyramid was developed and implemented, see [40]. We used the novel approach of accounting for orientations

at the level of the shape functions routine. In simple words, shape functions are constructed as restrictions of basis functions, so that assembling them into globally conforming basis functions becomes a trivial task.

- In collaboration with Kyungjoo Kim, a fellow graduate student, a refinement package was built from scratch. We have empirical evidence that the refinement scheme is deadlock-free. Constrained approximations allows computations on 1-irregular grids.
- The acoustics/elasticity coupled problem was successfully implemented and verified employing three test problems whose analytical solutions are known. This includes verification of the Perfectly Matched Layer as well.
- An explicit *a posteriori* error indicator for the coupled problem was developed and implemented. The implementation of the error indicator requires non-trivial integrations of jumps over the faces of possibly refined elements. The numerical experiments show that the error indicator exhibits the same behavior as the error in terms of rates of convergence, thus it is a reliable indicator.
- An extensive parametric study has been performed. We investigated the dependency upon frequency and direction of the incident pressure wave, as well as the effect of blocking the ear canal with an ear plug. Although we could not validate the numerical results against experimental medical data, they are consistent with the physical understanding of the hearing mechanism.

Future related research might include:

- A more realistic head model that includes details of a *real* cochlea and middle ear (ossicles and tympanic membrane) should be considered. A comparison with the results obtained for our *idealized* head model should assess how sensitive the problem is upon the geometry.

- The ossicles were crudely modeled as one elastic rod. A more sophisticated model that differentiate between the malleus, incus and stapes, as well as the way they are connected to each other, should be considered.

- The numerical results should be validated against experimental data. This would require an extensive review of the existing literature, as well as consulting with experts in the area of hearing physiology.

- The `hp3d` library could be further developed into different directions. On one hand, we could improve its flexibility and ease of use, in order to make more appealing to general users. Jeffry Zitelli, a fellow graduate student, has recently taken a lead in this direction by developing an interface with Python. On the other hand, `hp3d` could be extend to support non-conforming methods, e.g., the Discontinuous Petrov-Galerkin method developed by Demkowicz and Gopalakrishnan, see [24] and subsequent papers.

Appendices

Appendix A

G^1-Interpolation Scheme for Unstructured Triangular Grids

A.1 Scheme Construction

A.1.1 Twist Compatibility Condition

Second differentials, i.e., Hessians, as well as first differentials, i.e., normals, are the appropriate degrees of freedom on which the scheme is built. This fact is well established in the literature, see, e.g., [65]. Indeed, if we recall some elementary differential geometry, see, e.g., [29], and let $\mathbf{x} = \mathbf{x}(u_1, u_2)$ be a parameterization of a regular surface S, the Leibniz rule implies that

$$\langle dN_p(\mathbf{x}_{u_i}), \mathbf{x}_{u_j} \rangle = -\langle N_p, \mathbf{x}_{u_i u_j} \rangle \ ,$$

where the scalar product $\langle \cdot\,, \cdot \rangle$ is known as the first fundamental form and $dN_p : T_p(S) \to T_p(S)$ is the *Weingarten* map, i.e., the differential of the normal field N at $p \in S$. It is customary to define $II_p(u, v) = -\langle dN_p(u), v \rangle$, for $u, v \in T_p(S)$, and call it the second fundamental form. The above identity shows how the first and second order differentials are linked via the second fundamental form; in geometry reconstruction this fact is usually called *twist* compatibility condition. If we specialize the previous relation to $u = v$ we have that

$$II_p(v, v) = -\langle dN_p(v), v \rangle = \kappa_n \ .$$

In other words, the value of the second fundamental form II_p for a unit vector $v \in T_p(S)$ is equal to the normal curvature κ_n of a regular curve passing through p and tangent to v. Finally, if $\mathbf{n}$ is the normal and $A = \{a_{ij}\}$ is the Hessian of an implicit parameterization of the surface at the point, the twist compatibility condition can be rewritten as

$$\mathbf{x}_{u_i u_j} \cdot \mathbf{n} = -(A\,\mathbf{x}_{u_i}) \cdot \mathbf{x}_{u_j}\ . \tag{A.1.1}$$

This identity can be interpreted as the fact that, once the Hessian A and the first derivatives $\mathbf{x}_{u_i}$ have been determined, the *normal* component of $\mathbf{x}_{u_i u_j}$ has been determined as well.

A.1.2 Curves Network Reconstruction

The reconstruction of the curves network is performed as in [23], employing the reconstructed normals $\mathbf{n}$ and the Hessians $A = \{a_{ij}\}$ at the vertices. The normals and Hessians are computed by performing a least squares fit, and the curves that are generated by the reconstruction scheme are polynomials of degree 5. We give a quick overview of the whole procedure and refer to [23] for a detailed discussion.

At every vertex, the normal and the Hessian are reconstructed by minimizing the functional

$$L(A, \mathbf{n}) = \tfrac{1}{2} \sum_{l=1}^{N} w_l \left(\sum_{i,j=1}^{3} a_{ij} x_i^l x_j^l + \sum_{j=1}^{3} n_j x_j^l \right)^2 ,$$

under the constraint $\sum_{i=1}^{3} n_i^2 = 1$. N is the number of points surrounding the vertex—namely all the points within a fixed distance from the vertex—, w_l are some suitable weights, and $\mathbf{x}^l$ are the relative positions of the points with respect to the vertex.

The normal and Hessian are the stationary point of the Lagrangian $\mathcal{L}(A, \mathbf{n}, \mu) = L - \frac{1}{2}\mu\Big(\sum_i n_i^2 - 1\Big)$. The partial derivatives of $\mathcal{L}$ are:

$$\begin{aligned}
\frac{\partial \mathcal{L}}{\partial a_{mn}} &= \sum_{i,j=1}^{3} a_{ij}\left(\sum_l w_l x_i^l x_j^l x_m^l x_n^l\right) + \sum_{j=1}^{3} n_j\left(\sum_l w_l x_j^l x_m^l x_n^l\right) , \\
\frac{\partial \mathcal{L}}{\partial n_m} &= \sum_{i,j=1}^{3} a_{ij}\left(\sum_l w_l x_i^l x_j^l x_m^l\right) + \sum_{j=1}^{3} n_j\left(\sum_l w_l x_j^l x_n^l\right) - \mu n_m .
\end{aligned}$$

In order to expose the linear structure, we introduce a reordering $(i, j) : \{1, \ldots, 6\} \to \{1, 2, 3\}^2$ and define the matrices:

$$c_{IJ} = \sum_l w_l x_{i(I)}^l x_{j(I)}^l x_{m(J)}^l x_{n(J)}^l \quad ; \quad b_{Ij} = \sum_l w_l x_j^l x_{m(I)}^l x_{n(I)}^l \quad ; \quad d_{ij} = \sum_l w_l x_j^l x_i^l . \tag{A.1.2}$$

Since $\{a_{ij}\}$ is a symmetric matrix, its off-diagonal term must be counted twice, hence we have the cumbersome definition $a_J = \big(2 - \delta_{i(J)j(J)}\big) a_{i(J)j(J)}$. By setting the partial derivatives equal to zero we obtain the following linear system:

$$\begin{pmatrix} C & B \\ B^T & D \end{pmatrix} \begin{pmatrix} \boldsymbol{a} \\ \mathbf{n} \end{pmatrix} = \mu \begin{pmatrix} \mathbf{0} \\ \mathbf{n} \end{pmatrix} .$$

Upon eliminating $\boldsymbol{a}$ from the first equation, we end up with a standard eigenvalue problem for the unknown normal unit vector:

$$\big(D - B^T C^{-1} B\big)\mathbf{n} = \mu \mathbf{n} .$$

The eigenvalue μ coincides with the doubled value of the misfit function, consequently the best normal is the eigenvector corresponding to the smallest eigenvalue, see [23].

Let $\boldsymbol{\tau}$'s be the unit tangent vectors at a vertex. For a detailed discussion about determining those vectors refer to [23]. Let s be the curvilinear abscissa for a curve $\mathbf{x} = \mathbf{x}(t)$ and $\mathbf{n}_n$ be the *curve* principal normal. By employing a Frenet frame we have the well known formulas $\dot{\mathbf{x}} = \dot{s}\boldsymbol{\tau}$, $\ddot{\mathbf{x}} = \ddot{s}\boldsymbol{\tau} + \kappa_n(\dot{s})^2\mathbf{n}_n$. By assuming that the curve principal normal coincides, *modulo* orientation, with the surface normal, condition (A.1.1) implies that, at the vertices, the curve second derivative is equal to:

$$\ddot{\mathbf{x}} = \ddot{s}\boldsymbol{\tau} - (\dot{s})^2\big[(A\boldsymbol{\tau}) \cdot \boldsymbol{\tau}\big]\mathbf{n} \,.$$

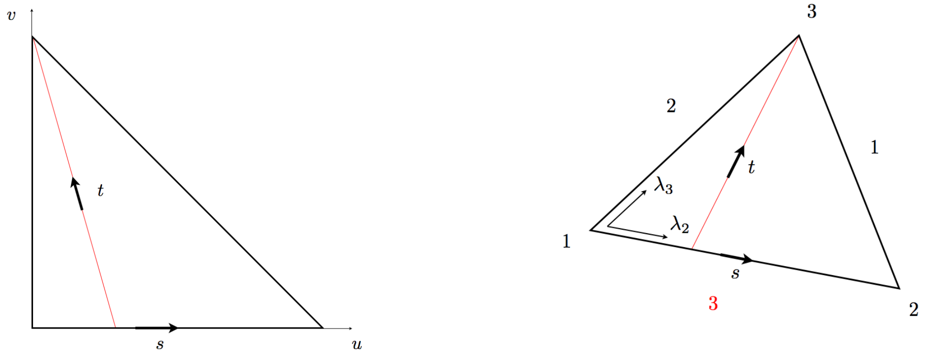
This fact motivates us to seek a reconstructed curve as:

$$\mathbf{x} = \varphi_1\mathbf{x}(0) + \varphi_2\mathbf{x}(1) + \varphi_3\dot{\mathbf{x}}(0) + \varphi_4\dot{\mathbf{x}}(1) + \varphi_5\ddot{\mathbf{x}}(0) + \varphi_6\ddot{\mathbf{x}}(1) \,, \tag{A.1.3}$$

where φ_i, for $j = 1, \ldots, 6$, are fifth order polynomials forming a dual basis to the degrees of freedom $\{\varphi(0), \varphi(1), \dot{\varphi}(0), \dot{\varphi}(1), \ddot{\varphi}(0), \ddot{\varphi}(1)\}$. The remaining degrees of freedom $\{\dot{s}(0), \dot{s}(1), \ddot{s}(0), \ddot{s}(1)\}$ are determined by minimizing the linearized curvature $\int_0^1 |\ddot{\mathbf{x}}| dt$, see [23] for further details.

A.1.3 G^1 Constraint

The tangent plane along an edge will be spanned by the curve velocity vector and by another vector that needs to be computed employing the vertex information only; this will guarantee that the scheme is completely local. Although the choice of such a vector is not unique, a good candidate is the directional derivative in the *radial* direction, since it matches the information available at the curve endpoints. Let us recall that the usual master triangle $\{0 \leq v \leq 1, 0 \leq u \leq 1 - v\}$ is the image of the master square $\{0 \leq s \leq 1, 0 \leq t \leq 1\}$ under the Duffy's transformation $u = s(1 - t)$, $v = t$. The Duffy's transformation can be interpreted as setting up a *moving* system



(a) Master triangle.

(b) Arbitrary triangle.

Figure A.1: Duffy's coordinates.

of coordinates along the bottom edge of the master triangle, see Figure A.1(a). By switching to barycentric coordinates, this system can be immediately derived for any edge of an arbitrary triangle. For example, the moving system along edge 3 of the triangle in Figure A.1(b) is described by $\lambda_2 = s(1-t)$, $\lambda_3 = t$, and, by cycling the indices, we obtain the desired construction for the remaining two edges. For a function $B = B(\lambda_i, \lambda_j)$ we set $\partial_{\text{rad}} B = \partial_t B\Big|_{t=0}$ and call this quantity the *radial* derivative of B along the j-th edge, according to the enumeration system shown in Figure A.1(b).

A.1.4 Bernstein Polynomials and Bezier Patches

The univaritate Berstein polynomials are defined as $B_i^n(t) = \binom{n}{i} t^i (1-t)^{n-i}$ for $i = 0, \ldots, n$ and $B_i^n(t) = 0$ when $i < 0$ or $i > n$. For $1 \leq i \leq n-1$ B_i^n is indeed a bubble on the interval $[0,1]$. Similarly, the bivariate Bernstein polynomials are defined as $B^n_{i,j}(u,v) = \frac{n!}{i!j!(n-i-j)!} u^i v^j (1-u-v)^{n-i-j}$ for $j = 0, \ldots, n$; $i = 0, \ldots, n-j$ and $B^n_{i,j}(u,v) = 0$ when $i < 0$ or $j < 0$ or $i+j > n$. Derivatives of the Berstein polynomials are easily computed using the following recursive relations:

$$\big(B^n_i\big)' = n\big(B^{n-1}_{i-1} - B^{n-1}_i\big)\,, \tag{A.1.4a}$$

$$\partial_u B^n_{i,j} = n\big(B^{n-1}_{i-1,j} - B^{n-1}_{i,j}\big)\,, \tag{A.1.4b}$$

$$\partial_v B^n_{i,j} = n\big(B^{n-1}_{i,j-1} - B^{n-1}_{i,j}\big)\,, \tag{A.1.4c}$$

$$\partial_{uv} B^n_{i,j} = n(n-1)\big(B^{n-2}_{i-1,j-1} - B^{n-2}_{i-1,j} - B^{n-2}_{i,j-1} + B^{n-2}_{i,j}\big)\,. \tag{A.1.4d}$$

Notice that by setting $u = \lambda_i$, $v = \lambda_j$ we can think of $B^n_{i,j}$ as a function of barycentric coordinates. By employing the Duffy's transformation $u = s(1-t)$, $v = t$, a bivariate Bernstein polynomial $B^n_{i,j}$ can be written as a product of univariate ones:

$$\begin{aligned}
B^n_{i,j}(u,v) &= \frac{n!}{i!j!(n-i-j)!} s^i (1-t)^i t^j (1-s)^{n-i-j} (1-t)^{n-i-j}\,, \\
&= \frac{(n-j)!}{i!\big((n-j)-i\big)!} s^i (1-s)^{(n-j)-i} \frac{n!}{j!(n-j)!} t^j (1-t)^{n-j}\,, \\
&= B^{n-j}_i(s)\, B^n_j(t)\,.
\end{aligned}$$

This result allows us to easily compute the radial derivative of a bivariate Bernstein polynomial along line $\{0 \le u \le 1, v = 0\}$. By recalling equation (A.1.4a) an the fact that $B^n_0(0) = 1$, $B^n_j(0) = 0$ for $j \ge 1$ we have that:

$$\partial_{\text{rad}} B^n_{i,j}(s) = (-)^{j+1}\, n\, B^{n-j}_i(s) \quad \text{for } j = 0,1 \qquad ; \qquad \partial_{\text{rad}} B^n_{i,j}(s) = 0 \quad \text{for } j \ge 2\,. \tag{A.1.5}$$

A triangular Bezier patch of degree n is defined *via* Bernstein polynomials as

$$\mathbf{P}_n(u,v) = \sum_{j=0}^{n} \sum_{i=0}^{n-j} \mathbf{b}_{ij}\, B^n_{i,j}(u,v)\,,$$

where $\mathbf{b}_{ij}$ are the control points. When equation (A.1.5) is applied to a triangular Bezier patch, we find that:

$$\partial_{\text{rad}}\mathbf{P}_n(s) = \sum_{j=0}^{n}\sum_{i=0}^{n-j} \mathbf{b}_{ij}\,\partial_{\text{rad}}B^n_{i,j}(s) = \sum_{j=0}^{1}\sum_{i=0}^{n-j} \mathbf{b}_{ij}\,(-)^{j+1}\,n\,B_i^{n-j}(s)\,.$$

The radial derivative of a triangular Bezier patch is fully determined by the first two rows of control points adjacent to the bottom edge, with obvious extensions to the remaining two edges. Indeed, it is a well known fact that the tangent plane along an edge of a triangular Bezier patch is fully determined by the two most adjacent rows of control points, see, *e.g.*, [35]. As a final remark, since we are trying to recreate a tensor product structure on the triangle through the Duffy's coordinates, the radial derivative is still a polynomial of degree n, and not $n-1$ as one could expect.

A.1.5 Computation of Radial Derivative

We compute the radial derivative in terms of the usual differentials $\partial_u\mathbf{P}$, $\partial_v\mathbf{P}$, $\partial^2_{uv}\mathbf{P}$, and so on, since this is the information that will be available—at least at the curve's endpoints! Let us consider a vertex shared by an *incoming* and an *outgoing* edge and set up a system of coordinates (u, v) as in Figure A.2(a). The radial derivative at the *start* point of the *outgoing* edge is computed by setting $u = s(1-t)$, $v = t$, see Figure A.2(b), and applying the chain rule:

$$\partial_t\mathbf{P} = \partial_u\mathbf{P}\,\partial_t u + \partial_v\mathbf{P}\,\partial_t v = -s\,\partial_u\mathbf{P} + \partial_v\mathbf{P}\,.$$

We compute $\partial^2_{st}\mathbf{P}$ in order to evaluate $\left(\partial_{\text{rad}}\mathbf{P}\right)'$ as well:

$$\begin{aligned}\partial^2_{st}\mathbf{P} &= \left[\partial_{uu}\mathbf{P}\,\partial_s u + \partial_{vu}\mathbf{P}\,\partial_s v\right]\partial_t u + \partial_u\mathbf{P}\,\partial^2_{st}u + \left[\partial_{uv}\mathbf{P}\,\partial_s u + \partial_{vv}\mathbf{P}\,\partial_s v\right]\partial_t v + \partial_v\mathbf{P}\,\partial^2_{st}v \\ &= s(t-1)\,\partial^2_{uu}\mathbf{P} - \partial_u\mathbf{P} + (1-t)\,\partial^2_{uv}\mathbf{P}\,.\end{aligned}$$

Similarly, the radial derivative at the *end* point of the *incoming* edge is computed by setting $u = t$, $v = (1 - s)(1 - t)$, see Figure A.2(c). The final results are:

$$\big(\partial_{\text{rad}}\mathbf{P}\big)_{\text{out}} = -s\partial_u\mathbf{P} + \partial_v\mathbf{P} \qquad \big(\partial_{\text{rad}}\mathbf{P}\big)'_{\text{out}} = -s\partial^2_{uu}\mathbf{P} - \partial_u\mathbf{P} + \partial^2_{uv}\mathbf{P}\,, \tag{A.1.6a}$$

$$\big(\partial_{\text{rad}}\mathbf{P}\big)_{\text{in}} = \partial_u\mathbf{P} + (s-1)\partial_v\mathbf{P} \qquad \big(\partial_{\text{rad}}\mathbf{P}\big)'_{\text{in}} = -(s-1)\partial^2_{vv}\mathbf{P} + \partial_v\mathbf{P} - \partial^2_{uv}\mathbf{P}\,. \tag{A.1.6b}$$

By evaluating the previous relations at the common vertex we find

$$\text{outgoing:} \qquad \partial_{\text{rad}}\mathbf{P}(0) = \partial_v\mathbf{P} \qquad , \qquad (\partial_{\text{rad}}\mathbf{P})'(0) = -\partial_u\mathbf{P} + \partial^2_{uv}\mathbf{P}\,; \tag{A.1.7a}$$

$$\text{incoming:} \qquad \partial_{\text{rad}}\mathbf{P}(1) = \partial_u\mathbf{P} \qquad , \qquad (\partial_{\text{rad}}\mathbf{P})'(1) = \partial_v\mathbf{P} - \partial^2_{uv}\mathbf{P}\,. \tag{A.1.7b}$$

Notice that, by picking a moving system of coordinates (s, t), the radial derivative coincides with the "right" single partial derivative at a curve's endpoints. On the other hand, the choice of a moving system of coordinates, results in the fact that $\big(\partial_{\text{rad}}\mathbf{P}\big)'_{\text{in}}(1)$ does not coincide with $\big(\partial_{\text{rad}}\mathbf{P}\big)'_{\text{out}}(0)$. This fact indicates that, on the implementation side, the mixed partial derivative $\partial^2_{uv}\mathbf{P}$ rather then $(\partial_{\text{rad}}\mathbf{P})'$ is the quantity of interest at the vertices, see Section A.2.

A.1.6 Radial Derivative Reconstruction

Let us consider a curve adjacent to a triangular patch, whose sought parameterization is $\mathbf{P}$, see Figure A.3. After the curves reconstruction step is carried out, the following quantities are known at the endpoints:

$$\partial_u\mathbf{P} \qquad , \qquad \partial_v\mathbf{P} \qquad , \qquad \partial^2_{uu}\mathbf{P} \qquad , \qquad \partial^2_{vv}\mathbf{P} \qquad , \qquad \partial^2_{uv}\mathbf{P}\cdot\mathbf{n}\,,$$

where indeed (u, v) refer to *different* systems of coordinates, according to whether we are considering the first or the second endpoint. According to the system of coordinates of the *first* endpoint, we seek a radial derivative of the form

$$\partial_{\text{rad}}\mathbf{P} = \alpha\,\partial_u\mathbf{P} + \beta\,\boldsymbol{\psi}\,, \tag{A.1.8}$$

where α, β, $\boldsymbol{\psi}$ are polynomial functions to be determined by the data at the endpoints. We pick $deg\,\beta = deg\,\boldsymbol{\psi} = 3$ in order to control *both* the value and the derivative of those functions at the endpoints. For simplicity we set $deg\,\alpha = 1$. In order to avoid confusion between the different systems of coordinates, we discuss conditions at the first endpoint only, and refer the reader to Section A.2 for full details about the second endpoint. We set

$$\boldsymbol{\psi}(0) = \frac{\mathbf{n}\times\partial_u\mathbf{P}}{|\mathbf{n}\times\partial_u\mathbf{P}|} \qquad ; \qquad \boldsymbol{\psi}'(0) = -\big[(A\,\partial_u\mathbf{P})\cdot\boldsymbol{\psi}\big]\mathbf{n}\,.$$

This choice implies that the tangential component of $\boldsymbol{\psi}'$ is set to zero. Hence $\boldsymbol{\psi}$ has been fully determined. Relation (A.1.8) is already satisfied at the endpoints in the normal direction since $\partial_{\text{rad}}\mathbf{P}$, $\partial_u\mathbf{P}$ and $\boldsymbol{\psi}$ are perpendicular to $\mathbf{n}$. We need to satisfy two independent conditions in the tangent plane only, which is achieved by setting

$$\alpha(0) = \frac{\partial_{\text{rad}}\mathbf{P}\cdot\partial_u\mathbf{P}}{|\partial_u\mathbf{P}|^2} = \frac{\partial_v\mathbf{P}\cdot\partial_u\mathbf{P}}{|\partial_u\mathbf{P}|^2} \qquad ; \qquad \beta(0) = \partial_{\text{rad}}\mathbf{P}\cdot\boldsymbol{\psi} = \partial_v\mathbf{P}\cdot\boldsymbol{\psi}\,.$$

The fact that $\{\partial_u\mathbf{P}, \boldsymbol{\psi}, \mathbf{n}\}$ is an orthogonal tern facilitates deriving those identities. In order to determine β', we compute the derivative of (A.1.8):

$$(\partial_{\text{rad}}\mathbf{P})' = \partial_u\partial_{\text{rad}}\mathbf{P} = \alpha'\,\partial_u\mathbf{P} + \alpha\,\partial^2_{uu}\mathbf{P} + \beta'\,\boldsymbol{\psi} + \beta\,\boldsymbol{\psi}'\,. \tag{A.1.9}$$

Such a relation is worth three independent scalar equations, while only one degree of freedom at the first endpoint is available. However, the twist compatibility condition

(A.1.1) implies that this identity is already satisfied in the normal direction. Indeed, working on the left hand side of (A.1.9), we have that:

$$\begin{aligned}\partial_u\partial_{\mathrm{rad}}\mathbf{P}\cdot\mathbf{n} = \partial^2_{uv}\mathbf{P}\cdot\mathbf{n} = -(A\,\partial_u\mathbf{P})\cdot\partial_v\mathbf{P} = -(A\,\partial_u\mathbf{P})\cdot\partial_{\mathrm{rad}}\mathbf{P} = \\ = -(A\,\partial_u\mathbf{P})\cdot(\alpha\,\partial_u\mathbf{P}+\beta\,\boldsymbol{\psi}) = -\alpha(A\,\partial_u\mathbf{P})\cdot\partial_u\mathbf{P} - \beta(A\,\partial_u\mathbf{P})\cdot\boldsymbol{\psi}\,.\end{aligned}$$

Similarly, working on the right hand side and recalling the definition of $\boldsymbol{\psi}'$, we obtain

$$\begin{aligned}(\alpha'\,\partial_u\mathbf{P}+\alpha\,\partial^2_{uu}\mathbf{P}+\beta'\,\boldsymbol{\psi}+\beta\,\boldsymbol{\psi}')\cdot\mathbf{n} = \\ = \alpha\,\partial^2_{uu}\mathbf{P}\cdot\mathbf{n}+\beta\,\boldsymbol{\psi}'\cdot\mathbf{n} = -\alpha(A\,\partial_u\mathbf{P})\cdot\partial_u\mathbf{P}-\beta(A\,\partial_u\mathbf{P})\cdot\boldsymbol{\psi}\,.\end{aligned}$$

Therefore condition (A.1.9) needs to be enforced in the tangent plane only. This can be done since, beside β', we are still free to choose the tangential component of $\partial_u\partial_{\mathrm{rad}}\mathbf{P}$. Compatibility in the $\partial_u\mathbf{P}$ direction is immediately achieved by dotting (A.1.9) with $\partial_u\mathbf{P}$ and recalling that the tangential component of $\boldsymbol{\psi}'$ was set to zero:

$$\partial_u\partial_{\mathrm{rad}}\mathbf{P}\cdot\partial_u\mathbf{P} = \alpha'\,|\partial_u\mathbf{P}|^2+\alpha\,\partial^2_{uu}\mathbf{P}\cdot\partial_u\mathbf{P}\,.$$

By considering the next counterclock-wise curve in the bundle originating from the curve first endpoint, we determine $\partial_u\partial_{\mathrm{rad}}\mathbf{P}\cdot\partial_v\mathbf{P}$. At this point $\partial_u\partial_{\mathrm{rad}}\mathbf{P}$ is fully known. Finally, if we dot equation (A.1.9) by $\boldsymbol{\psi}$, we obtain an expression for $\beta'(0)$:

$$\beta'(0) = (\partial_u\partial_{\mathrm{rad}}\mathbf{P}-\alpha'\,\partial_u\mathbf{P}-\alpha\,\partial^2_{uu}\mathbf{P})\cdot\boldsymbol{\psi}\,, \tag{A.1.10}$$

which insures compatibility in the $\boldsymbol{\psi}$ direction.

A.1.7 Patch Reconstruction

Since the radial derivative has been reconstructed as a polynomial of degree 6, this would suggest employing a Bezier patch of degree 6. Notice, however, that in the case of a sextic Bezier triangle, see Figure A.4(a), only 6 control points are adjacent to each edge. Let us suppose for a second that the parameterization has already been fixed on the bottom edge only. We are left with 6 control points, namely $\{\mathbf{b}_{01}, \mathbf{b}_{11}, \mathbf{b}_{21}, \mathbf{b}_{31}, \mathbf{b}_{41}, \mathbf{b}_{51}\}$, for matching the radial derivative along that edge. *In general* this will not work, because of the trivial fact that seven dof's are needed to match a degree 6 polynomial. Unfortunately, we need to employ Bezier patches of degree 7, see Figure A.4(b). The points on the boundary, shown in black, are determined by the curves reconstruction scheme. The control points shown in red depend upon the radial derivatives along the edges. Let's focus on the bottom edge again. Indeed control points $\mathbf{b}_{01}, \mathbf{b}_{61}$ have already been set, but the twist compatibility condition ensures that they indeed match the radial derivative at the curve endpoints. A similar reasoning applies to the corner control points $\mathbf{b}_{11}, \mathbf{b}_{51}, \mathbf{b}_{15}$ which are common to the radial derivatives along two edges. Full details about how to determine the control points are given in the next section. We conclude by noticing that we are left with three inner control points, shown in blue, that do not affect the G^1 constraint along the edges. A number of strategies can be used to set those control points. They can be set in order to get more pleasing shapes, or they can be set to quintic precision, in order to leave open to possibility of recovering a degree 5 patch. Another simple solution is to set them to some average of the surrounding control points.

A.2 Scheme Implementation

A.2.1 Wireframe Reconstruction

Given and unstructured linear triangular mesh, i.e., a set of vertices and their connectivities, the curves network is reconstructed through the following steps:

Step 1 For each vertex, determine the normal $\mathbf{n}$ by selecting the eigenvector associated to the smallest eigenvalue of the problem

$$\big(D - B^T C^{-1} B\big)\mathbf{n} = \mu \mathbf{n}\,.$$

where matrices B, C, D are defined in (A.1.2). The Hessian $A = \{a_{ij}\}$ is determined as $\boldsymbol{a} = -C^{-1}B\mathbf{n}$. Matrix C is singular when the least square fit is performed employing less then six points. A solution to this problem is illustrated in Section 4.1 of [23].

Step 2 For each vertex set up tangent unit vectors $\boldsymbol{\tau}$'s using the strategy described in Section 2.3 of [23].

Step 3 For each curve, seek a parameterization $\mathbf{x}$ of the form (A.1.3). At the endpoints the following relations hold:

$$\dot{\mathbf{x}} = \dot{s}\boldsymbol{\tau} \qquad ; \qquad \ddot{\mathbf{x}} = \ddot{s}\boldsymbol{\tau} - (\dot{s})^2\big[(A\boldsymbol{\tau})\cdot\boldsymbol{\tau}\big]\mathbf{n}\,.$$

Determine the unknowns $\dot{s}(0), \dot{s}(1), \ddot{s}(0), \ddot{s}(1)$ by minimizing the linearized curvature $\int_0^1 |\ddot{\mathbf{x}}|^2$. Notice that, due to the quadratic dependence of the second derivative at the endpoints upon the first derivatives $\dot{s}$, the resulting system of algebraic equations is nonlinear and has to be solved using, e.g., Newton-Raphson iterations.

A.2.2 Patch Reconstruction

The following identity relating control points to the mixed partial derivative of a Bezier triangle is well-known:

$$\partial^2_{uv}\mathbf{P}_n(0,0) = n(n-1)\big(\mathbf{b}_{11} - \mathbf{b}_{10} - \mathbf{b}_{01} + \mathbf{b}_{00}\big)\,. \tag{A.2.11}$$

In other words, the mixed derivative at the vertices determines the corner control points $\mathbf{b}_{11}$, $\mathbf{b}_{(n-2)1}$, and $\mathbf{b}_{1(n-2)}$. We wish to derive similar relations for the radial derivative. Employing equation (A.1.4a), this is indeed an elementary task:

$$\begin{aligned}
\big(\partial_{\mathrm{rad}}\mathbf{P}_n\big)^{(0)} &= \sum_{j=0}^{1}\sum_{i=0}^{n-j} \mathbf{b}_{ij}\,(-)^{j+1}\,n\,B_i^{n-j}\,,\\
\big(\partial_{\mathrm{rad}}\mathbf{P}_n\big)^{(1)} &= \sum_{j=0}^{1}\sum_{i=0}^{n-j} \mathbf{b}_{ij}\,(-)^{j+1}\,n(n-j)\big(B_{i-1}^{n-j-1} - B_i^{n-j-1}\big)\,,\\
\big(\partial_{\mathrm{rad}}\mathbf{P}_n\big)^{(2)} &= \sum_{j=0}^{1}\sum_{i=0}^{n-j} \mathbf{b}_{ij}\,(-)^{j+1}\,n(n-j)(n-j-1)\big(B_{i-2}^{n-j-2} - 2B_{i-1}^{n-j-2} + B_i^{n-j-2}\big)\,,\\
\big(\partial_{\mathrm{rad}}\mathbf{P}_n\big)^{(3)} &= \sum_{j=0}^{1}\sum_{i=0}^{n-j} \mathbf{b}_{ij}\,(-)^{j+1}\,n(n-j)(n-j-1)(n-j-2)\\
&\qquad\qquad \times\big(B_{i-3}^{n-j-3} - 3B_{i-2}^{n-j-3} + 3B_{i-1}^{n-j-3} + B_i^{n-j-3}\big)\,.
\end{aligned}$$

By letting m be the order of derivation and defining coefficients $c_{ij}(m,n,s)$ appropriately, the previous four identities are reduced to a single one:

$$\big(\partial_{\mathrm{rad}}\mathbf{P}_n\big)^{(m)}(s) = \sum_{j=0}^{1}\sum_{i=0}^{n-j} \mathbf{b}_{ij}\,c_{ij}(m,n,s)\,.$$

Evaluating the previous relation at $s = 0, 1$ and introducing the auxiliary index $k = n-i-j$, we obtain the important result:

$$\big(\partial_{\mathrm{rad}}\mathbf{P}_n\big)^{(m)}(0) = \sum_{j=0}^{1}\sum_{i=0}^{m} \mathbf{b}_{ij}\,c_{ij}(m,n,0) \quad ; \quad \big(\partial_{\mathrm{rad}}\mathbf{P}_n\big)^{(m)}(1) = \sum_{j=0}^{1}\sum_{k=0}^{m} \mathbf{b}_{ij}\,c_{kj}(m,n,1)\,. \tag{A.2.12}$$

The radial derivative is fully determined by the information at the endpoints, which is available after the wireframe reconstruction is performed. This implies that its derivatives, in particular the second and third derivatives, which are

$$\big(\partial_{\mathrm{rad}}\mathbf{P}\big)^{(2)} = 2\alpha'\partial^2_{uu}\mathbf{P} + \alpha\partial^3_{uuu}\mathbf{P} + \beta''\boldsymbol{\psi} + 2\beta'\boldsymbol{\psi}' + \beta\boldsymbol{\psi}'' \,, \tag{A.2.13a}$$

$$\big(\partial_{\mathrm{rad}}\mathbf{P}\big)^{(3)} = 3\alpha'\partial^3_{uuu}\mathbf{P} + \alpha\partial^4_{uuuu}\mathbf{P} + \beta'''\boldsymbol{\psi} + 3\beta''\boldsymbol{\psi}' + 3\beta'\boldsymbol{\psi}'' + \beta\boldsymbol{\psi}''' \tag{A.2.13b}$$

are known as well. In particular, equations (A.2.13) and (A.2.12) allow us to work *forward* in order to determine $\mathbf{b}_{21}$ and $\mathbf{b}_{31}$, and work *backward* to determine $\mathbf{b}_{41}$. The same ideas are employed to determine $\{\mathbf{b}_{42}, \mathbf{b}_{33}, \mathbf{b}_{24}\}$ and $\{\mathbf{b}_{14}, \mathbf{b}_{13}, \mathbf{b}_{12}\}$.

Let us fix some notation in order to clearly describe the triangle reconstruction scheme. Let's introduce a local enumeration of the triangle vertices. The i-th edge of the triangle is oriented from the i-th to the $(i+1)$-th vertex. All indices are understood *modulo* 3. At each vertex i we introduce a system of coordinates (u_i, v_i)– with u_i going from vertex i to vertex $i+1$, and v_i going from vertex i to vertex $i+2$–, and *exclusively* use this system when talking about a differential quantity computed at the vertex. Finally, for each vertex we define vectors $(\mathbf{u}^i, \mathbf{v}^i)$ forming a dual basis with respect to vectors $(\partial_{u_i}\mathbf{P}, \partial_{v_i}\mathbf{P})$, so that

$$\partial^2_{u_iv_i}\mathbf{P} = \big(\partial^2_{u_iv_i}\mathbf{P}\cdot\partial_{u_i}\mathbf{P}\big)\mathbf{u}^i + \big(\partial^2_{u_iv_i}\mathbf{P}\cdot\partial_{v_i}\mathbf{P}\big)\mathbf{v}^i + \big(\partial^2_{u_iv_i}\mathbf{P}\cdot\mathbf{n}_i\big)\mathbf{n}_i \,.$$

By combining together equation (A.1.9) and equations (A.1.7) we have that

$$\partial^2_{u_iv_i}\mathbf{P} = \big(\alpha'(0)+1\big)\partial_{u_i}\mathbf{P} + \alpha(0)\partial^2_{u_iu_i}\mathbf{P} + \beta'(0)\boldsymbol{\psi}(0) + \beta(0)\boldsymbol{\psi}'(0) \,,$$

$$\partial^2_{u_{i+1}v_{i+1}}\mathbf{P} = \big(\alpha'(1)+1\big)\partial_{v_{i+1}}\mathbf{P} - \alpha(1)\partial^2_{v_{i+1}v_{i+1}}\mathbf{P} - \beta'(1)\boldsymbol{\psi}(1) - \beta(1)\boldsymbol{\psi}'(1) \,.$$

where α, β, $\boldsymbol{\psi}$ refer to the i-th edge. This implies that:

$$\partial^2_{u_i v_i}\mathbf{P} \cdot \partial_{u_i}\mathbf{P} = \big(\alpha'(0)+1\big)|\partial_{u_i}\mathbf{P}|^2 + \alpha(0)\partial^2_{u_i u_i}\mathbf{P} \cdot \partial_{u_i}\mathbf{P}\ , \tag{A.2.15a}$$

$$\partial^2_{u_{i+1} v_{i+1}}\mathbf{P} \cdot \partial_{v_{i+1}}\mathbf{P} = \big(\alpha'(1)+1\big)|\partial_{v_{i+1}}\mathbf{P}|^2 - \alpha(1)\partial^2_{v_{i+1} v_{i+1}}\mathbf{P} \cdot \partial_{v_{i+1}}\mathbf{P}\ . \tag{A.2.15b}$$

The reconstruction procedure for a *fixed* triangle is performed as follows.

Step 1 For each vertex initialize the mixed derivative to the normal component:

$$\partial^2_{u_i v_i}\mathbf{P} = \big(\partial^2_{u_i v_i}\mathbf{P} \cdot \mathbf{n}_i\big)\mathbf{n}_i = -\big[(A\partial_{u_i}\mathbf{P}) \cdot \partial_{v_i}\mathbf{P}\big]\mathbf{n}_i\ .$$

Step 2 For each edge compute:

$$\begin{aligned}
\alpha(0) &= \frac{\partial_{v_i}\mathbf{P} \cdot \partial_{u_i}\mathbf{P}}{|\partial_{u_i}\mathbf{P}|^2} & \alpha(1) &= -\frac{\partial_{v_{i+1}}\mathbf{P} \cdot \partial_{u_{i+1}}\mathbf{P}}{|\partial_{v_{i+1}}\mathbf{P}|^2}\ ,\\
\boldsymbol{\psi}(0) &= \frac{\mathbf{n}_i \times \partial_{u_i}\mathbf{P}}{|\mathbf{n}_i \times \partial_{u_i}\mathbf{P}|} & \boldsymbol{\psi}(1) &= -\frac{\mathbf{n}_{i+1} \times \partial_{v_{i+1}}\mathbf{P}}{|\mathbf{n}_{i+1} \times \partial_{v_{i+1}}\mathbf{P}|}\ ,\\
\beta(0) &= \partial_{v_i}\mathbf{P} \cdot \boldsymbol{\psi}(0) & \beta(1) &= \partial_{u_{i+1}}\mathbf{P} \cdot \boldsymbol{\psi}(1)\ .
\end{aligned}$$

Step 3 For each edge compute contributions to the mixed derivative at the curve endpoints in the $\mathbf{u}^i$ and $\mathbf{v}^{i+1}$ direction respectively, employing equations (A.2.15). Update the mixed derivative:

$$\begin{aligned}
\partial^2_{u_i v_i}\mathbf{P} &\leftarrow \partial^2_{u_i v_i}\mathbf{P} + \big(\partial^2_{u_i v_i}\mathbf{P} \cdot \partial_{u_i}\mathbf{P}\big)\mathbf{u}^i\ ,\\
\partial^2_{u_{i+1} v_{i+1}}\mathbf{P} &\leftarrow \partial^2_{u_{i+1} v_{i+1}}\mathbf{P} + \big(\partial^2_{u_{i+1} v_{i+1}}\mathbf{P} \cdot \partial_{v_{i+1}}\mathbf{P}\big)\mathbf{v}^{i+1}\ .
\end{aligned}$$

At this point $\partial^2_{u_i v_i}\mathbf{P}$ is fully determined for each triangle vertex.

Step 4 For each edge compute $\big(\partial_{\text{rad}}\mathbf{P}\big)^{(1)}$ at the curve's endpoints by adapting equations (A.1.7), *i.e.*:

$$\big(\partial_{\text{rad}}\mathbf{P}\big)^{(1)}(0) = -\partial_{u_i}\mathbf{P} + \partial^2_{u_i v_i}\mathbf{P} \qquad ; \qquad \big(\partial_{\text{rad}}\mathbf{P}\big)^{(1)}(1) = \partial_{v_{i+1}}\mathbf{P} - \partial^2_{u_{i+1}v_{i+i}}\mathbf{P} \,,$$

determine $\beta'(0)$ and $\beta'(1)$ by adapting equation (A.1.10):

$$\beta'(0) = \Big(\big(\partial_{\text{rad}}\mathbf{P}\big)^{(1)}(0) - \alpha'(0)\partial_{u_i}\mathbf{P} - \alpha(0)\partial^2_{u_i u_i}\mathbf{P}\Big) \cdot \boldsymbol{\psi}(0) \,.$$
$$\beta'(1) = \Big(\big(\partial_{\text{rad}}\mathbf{P}\big)^{(1)}(1) + \alpha'(1)\partial_{v_{i+1}}\mathbf{P} - \alpha(1)\partial^2_{v_{i+1}v_{i+1}}\mathbf{P}\Big) \cdot \boldsymbol{\psi}(1) \,.$$

At this point the radial derivative along each edge and, consequently, all of its derivatives are fully determined.

Step 5 Determine control points $\mathbf{b}_{11}$, $\mathbf{b}_{51}$, $\mathbf{b}_{15}$ using equation (A.2.11).

Step 6 For each edge compute $\big(\partial_{\text{rad}}\mathbf{P}\big)^{(2)}$ and $\big(\partial_{\text{rad}}\mathbf{P}\big)^{(3)}$ using (A.2.13). By evaluating such expressions at 0 or 1, set control points $\mathbf{b}_{21}$, $\mathbf{b}_{31}$, $\mathbf{b}_{41}$ using equation (A.2.12).

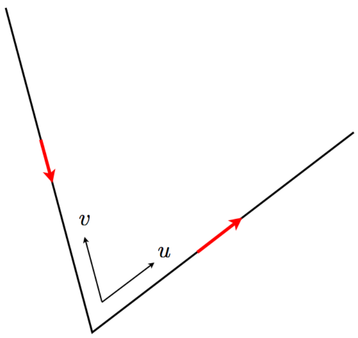
Step 7 Set $\mathbf{b}_{22}$, $\mathbf{b}_{32}$, $\mathbf{b}_{23}$ to the average of their surrounding control points, for example:

$$\mathbf{b}_{22} = \frac{\mathbf{b}_{31} + \mathbf{b}_{13}}{2} \qquad , \qquad \mathbf{b}_{32} = \frac{\mathbf{b}_{31} + \mathbf{b}_{33}}{2} \qquad , \qquad \mathbf{b}_{23} = \frac{\mathbf{b}_{33} + \mathbf{b}_{13}}{2} \,.$$

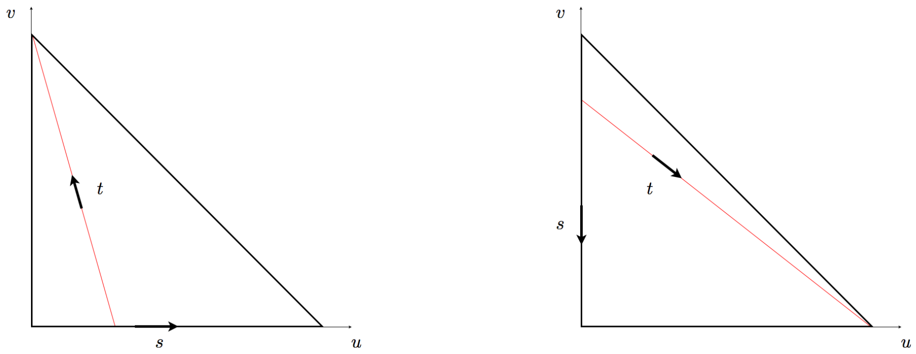
Indeed those control points do not affect the G^1 constraint along the edges.

A.2.3 Scheme Testing

A human head phantom coming from COMSOL®, a commercial FE code, was employed for testing our scheme. A faceted representation with 14316 triangles of the phantom was extracted from COMSOL®, see Figure A.5(a) and input into NETGEN, see [70], to obtain a coarse mesh of 538 triangles, see Figure A.5(b). The faceted surface was used as fine grid information for computing the normals and the Hessians at the vertices: for each vertex a least square fit over all points within a distance of 0.015 m was performed. As can be seen from the wireframe in Figure A.5(c), sharp edges, i.e., edges along which the G^1 continuity constraint should not be enforced, were detected on the lower part of the phantom. Sharp edges are treated as regular ones, with the *caveat* that for every vertex two normals and Hessians were reconstructed, using information of one adjacent surface at the time. The final result of the reconstruction scheme is shown in Figure A.5(d); we verified that the reconstructed phantom is indeed a G^1 object, up to machine precision, by sampling the normal at 10 points along each edge.



(a) (u, v)-coordinates at the vertex.



(b) (s, t)-coordinates on the outgoing edge.

(c) (s, t)-coordinates on the incoming edge.

Figure A.2: Moving systems of coordinates along edges.

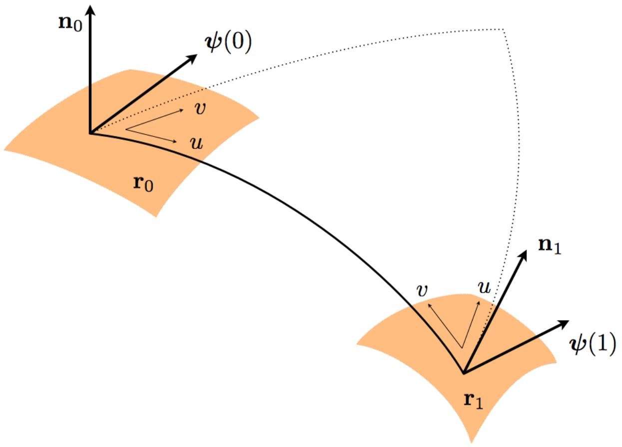
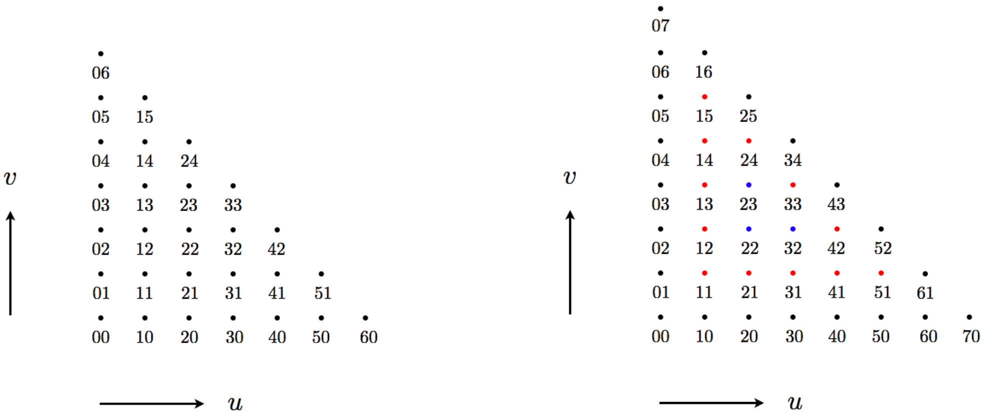


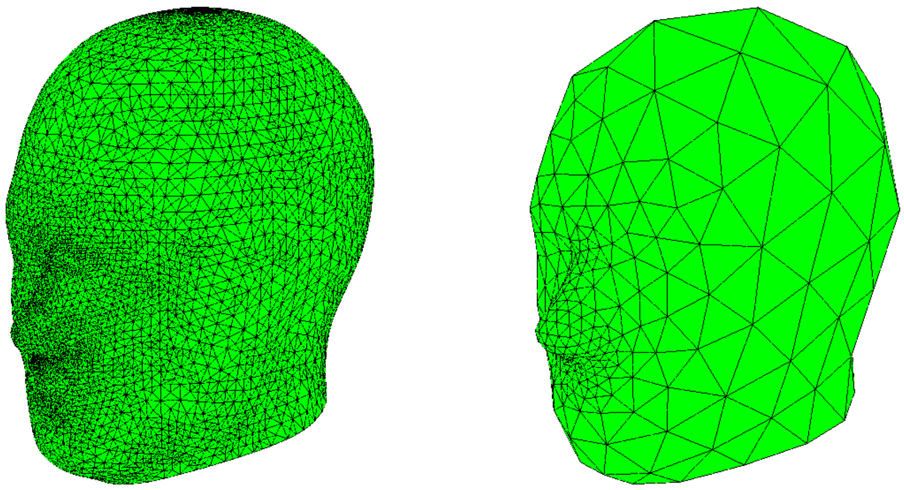
Figure A.3: Radial derivative reconstruction.



(a) Sextic Bezier triangle.

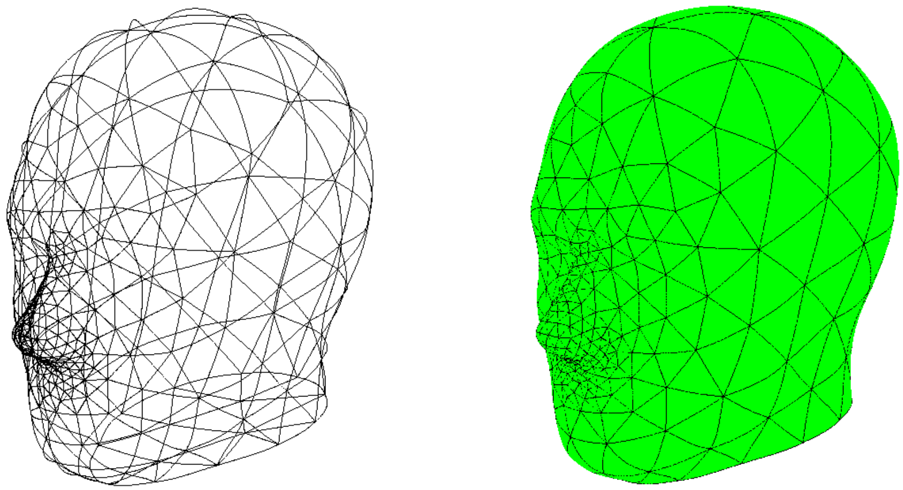
(b) Septic Bezier triangle.

Figure A.4: Control points of Bezier triangles.



(a) COMSOL® fine grid.

(b) NETGEN coarse mesh.



(c) Reconstructed wireframe.

(d) Reconstructed surface.

Figure A.5: Head phantom reconstruction.

Bibliography

[1] Mark Ainsworth and Joe Coyle. Hierarchic finite element bases on unstructured tetrahedral meshes. *International Journal for Numerical Methods in Engineering*, 58(14):2103–2130, 2003.

[2] Mark Ainsworth and J. Tinsley Oden. A posteriori error estimation in finite element analysis. *Computer Methods in Applied Mechanics and Engineering*, 142(1-2):1–88, 1997.

[3] Patrick R. Amestoy, Abdou Guermouche, Jean-Yves L'Excellent, and Stéphane Pralet. Hybrid scheduling for the parallel solution of linear systems. *Parallel Comput.*, 32(2):136–156, February 2006.

[4] Jean-Pierre Berenger. A perfectly matched layer for the absorption of electromagnetic waves. *J. Comput. Phys.*, 114(2):185–200, October 1994.

[5] C. Bernardi and V. Girault. A local regularization operator for triangular and quadrilateral finite elements. *SIAM Journal on Numerical Analysis*, 35(5):pp. 1893–1916, 1998.

[6] R.T. Beyer. *Nonlinear Acoustics*. Acoustical Society of America, 1st edition, 1997.

[7] Paolo Bientinesi, Victor Eijkhout, Kyungjoo Kim, Jason Kurtz, and Robert van de Geijn. Sparse direct factorizations through unassembled hyper-matrices.

Computer Methods in Applied Mechanics and Engineering, 199(9-12):430–438, 2010.

[8] D.T. Blackstock. *Fundamentals of Physical Acoustics.* Wiley-Interscience, 2000.

[9] F. Böhnke and W. Arnold. Bone conduction in a three-dimensional model of the cochlea. *Journal of Oto-Rhino-Laryngology*, 68(6):393–396, 2006.

[10] J.D. Boissonnat. *Effective Computational Geometry for Curves and Surfaces.* Springer, 1st edition, 2006.

[11] J.H. Bramble and J.E. Pasciak. Analysis of a finite pml approximation for the three dimensional time-harmonic Maxwell and acoustic scattering problems. *Mathematics of Computations*, 76:597–614, 2007.

[12] Annalisa Buffa. Remarks on the discretization of some noncoercive operator with applications to heterogeneous Maxwell equations. *SIAM Journal on Numerical Analysis*, 43(1):pp. 1–18, 2006.

[13] G.F. Carey. *Computational Grids: Generations, Adaptation, and Solution Strategies.* CRC Press, 1st edition, 1997.

[14] M. Chafi, G. Karami, and M. Ziejewski. Biomechanical assessment of brain dynamic responses due to blast pressure waves. *Annals of Biomedical Engineering*, 38:490–504, 2010.

[15] Y. C. Chang and L. Demkowicz. Solution of viscoelastic scattering problems in linear acoustics using *hp* boundary/finite element method. *International Journal for Numerical Methods in Engineering*, 44(12):1885–1907, 1999.

[16] Xiangjiu Che, Xuezhang Liang, and Qiang Li. G^1 continuity conditions of adjacent nurbs surfaces. *Computer Aided Geometric Design*, 22(4):285–298, 2005.

[17] Weng Cho Chew and William H. Weedon. A 3d perfectly matched medium from modified Maxwell's equations with stretched coordinates. *Microwave and Optical Technology Letters*, 7(13):599–604, 1994.

[18] Francis Collino and Peter Monk. The perfectly matched layer in curvilinear coordinates. *SIAM J. Sci. Comput.*, 19(6):2061–2090, November 1998.

[19] L. Demkowicz. Asymptotic convergence in finite and boundary element methods. Part 1: theoretical results. *Computers & Mathematics with Applications*, 27(12):69–84, 1994.

[20] L. Demkowicz. Asymptotic convergence in finite and boundary element methods. Part 2: the LBB constant for rigid and elastic scattering problems. *Computers & Mathematics with Applications*, 28(6):93–109, 1994.

[21] L. Demkowicz. *Computing with hp-Finite Elements. Volume I. One- and Two-Dimensional Elliptic and Maxwell Problems.* Chapman & Hall/CRC Press, 1st edition, 2006.

[22] L. Demkowicz, P. Gatto, J. Kurtz, M. Paszyński, W. Rachowicz, E. Bleszyński, M. Bleszyński, M. Hamilton, C. Champlin, and D. Pardo. Modeling of bone conduction of sound in the human head using hp-finite elements: Code design and verification. *Computer Methods in Applied Mechanics and Engineering*, 200(21-22):1757–1773, 2011.

[23] L. Demkowicz, P. Gatto, W. Qiu, and A. Joplin. G^1-interpolation and geometry reconstruction for higher order finite elements. *Computer Methods in Applied Mechanics and Engineering*, 198(13-14):1198 – 1212, 2009.

[24] L. Demkowicz and J. Gopalakrishnan. A class of discontinuous Petrov-Galerkin methods. part i: The transport equation. *Computer Methods in Applied Mechanics and Engineering*, 199(2324):1558 – 1572, 2010.

[25] L. Demkowicz, J. Kurtz, D. Pardo, M. Paszyński, W. Rachowicz, and A. Zdunek. *Computing with hp-Finite Elements. Volume II. Frontiers: Three-Dimensional Elliptic and Maxwell Problems with Applications.* Chapman & Hall/CRC Press, 1st edition, 2007.

[26] L. Demkowicz and J.T. Oden. Application of *hp*-adaptive be/fe methods to elastic scattering. *Computer Methods in Applied Mechanics and Engineering*, 133(3-4):287–317, 1996.

[27] Leszek Demkowicz. Polynomial exact sequences and projection-based interpolation with application to Maxwell equations. In *Mixed Finite Elements, Compatibility Conditions, and Applications*, volume 1939 of *Lecture Notes in Mathematics*, pages 101–158. Springer Berlin / Heidelberg, 2008.

[28] Leszek Demkowicz, Jason Kurtz, and Frederick Qiu. *hp*-adaptive finite elements for coupled multiphysics wave propagation problems. In M. Kuczma and K. Wilmański, editors, *Computer Methods in Mechanics*, volume 1 of *Advanced Structured Materials*, pages 19–42. Springer Berlin Heidelberg, 2010.

[29] M.P. do Carmo. *Differential Geometry of Curves and Surfaces.* Prentice-Hall, 1st edition, 1976.

[30] Moshe Dubiner. Spectral methods on triangles and other domains. *Journal of Scientific Computing*, 6:345–390, 1991.

[31] G. Duvaut and J.L. Lions. *Les Inèquations en Mécanique et en Physique.* Dunod, Paris, 1972.

[32] A.C. Eringen and E.S. Suhubi. *Elastodynamics.* Academic Press, 1975.

[33] Ph. Devloo et al. Systematic and generic construction of shape functions for *p*-adaptive meshes of multidimensional finite elements. *Computer Methods in Applied Mechanics and Engineering*, 198(21-26):1716–1725, 2009. Advances in Simulation-Based Engineering Sciences - Honoring J. Tinsley Oden.

[34] L.C. Evans. *Partial Differential Equations.* American Mathematical Society, 1998.

[35] G. Farin. *Curves and Surfaces for CADG.* Morgan-Kaufmann, 5th edition, 2001.

[36] Rong Gan, Brian Reeves, and Xuelin Wang. Modeling of sound transmission from ear canal to cochlea. *Annals of Biomedical Engineering*, 35:2180–2195, 2007.

[37] Rong Z. Gan, Qunli Sun, Bin Feng, and Mark W. Wood. Acoustic-structural coupled finite element analysis for sound transmission in human ear—pressure distributions. *Medical Engineering & Physics*, 28(5):395–404, 2006.

[38] S. Ganpule, L. Gu, G. Cao, and N. Chandra. The effect of shock wave on a human head. In *Proceedings of IMECE 2009*, November 2009.

[39] G.N. Gatica, A. Márquez, and S. Meddahi. Analysis of the coupling of primal and dual-mixed finite element methods for a two-dimensional fluid-solid interaction problem. *SIAM J. Numer. Anal.*, 45(5):2072–2097, August 2007.

[40] P. Gatto and L. Demkowicz. Construction of H^1-conforming hierarchical shape functions for elements of all shapes and transfinite interpolation. *Finite Elem. Anal. Des.*, 46(6):474–486, June 2010.

[41] K. Gerdes and L. Demkowicz. Solution of 3d-Laplace and Helmholtz equations in exterior domains using *hp*-infinite elements. *Computer Methods in Applied Mechanics and Engineering*, 137(3-4):239–273, 1996.

[42] E. Goodman. *Handbook of Discrete and Computational Geometry.* Chapman and Hall/CRC, 2nd edition, 2004.

[43] M.F. Hamilton and D.T. Blackstock. *Nonlinear Acoustics.* Academic Press, 1st edition, 1997.

[44] J. Sanchez Hubert and E. Sanchez Palencia. *Vibration and Coupling of Continuous Systems: Asymptotic Methods.* Springer, 1989.

[45] Christian James. Finite element modeling and exploration of double hearing protection systems. Master's thesis, Virginia Polytechnic Institute and State University, 2006.

[46] P. Jiránek, Z. Strakoš, and M. Vohralík. A posteriori error estimates including algebraic error and stopping criteria for iterative solvers. *SIAM Journal on Scientific Computing*, 32(3):1567–1590, 2010.

[47] E.A.C. Johnson and P.G. Young. On the use of a patient-specific rapid-prototyped model to simulate the response of the human head to impact and comparison with analytical and finite element models. *Journal of Biomechanics*, 38(1):39–45, 2005.

[48] W.D. O'Brien Jr. and Y. Liu. Evaluation of acoustic propagation paths into the human head. In *New Directions for Improving Audio Effectiveness*, pages 15–1 – 15–24, 2005.

[49] G.E. Karniadakis and S.J. Sherwin. *Spectral/hp Element Methods for CFD.* Oxford University Press, 1999.

[50] L.D. Landau and E.M. Lifshitz. *Fluid Mechanics.* Butterworth-Heinemann, 2nd edition, 1987.

[51] L.D. Landau and E.M. Lifshitz. *Theory of Elasticity.* Butterworth-Heinemann, 2nd edition, 1987.

[52] M. Lassas and E. Somersalo. On the existence and convergence of the solution of PML equations. *Computing*, 60:229–241, 1998.

[53] Matti Lassas and Erkki Somersalo. Analysis of the PML equations in general convex geometry. *Proceedings of the Royal Society of Edinburgh, Section: A Mathematics*, 131(05):1183–1207, 2001.

[54] Rolf Leis. *Initial boundary value problems in mathematical physics.* McGraw-Hill, 1986.

[55] Charles Loop. A G^1 triangular spline surface of arbitrary topological type. *Computer Aided Geometric Design*, 11(3):303–330, 1994.

[56] A. Majda. *Compressible Fluid Flow and Systems of Conservation Laws in Several Space Variables.* Springer-Verlag, 1st edition, 1984.

[57] P.J. Matuszyk, L.F. Demkowicz, and C. Torres-Verdin. Solution of coupled acoustic-elastic wave propagation problems with anelastic attenuation using automatic *hp*-adaptivity. *Computer Methods in Applied Mechanics and Engineering*, 213–216(0):299–313, 2012.

[58] C. Michler, L. Demkowicz, J. Kurtz, and D. Pardo. Improving the performance of perfectly matched layers by means of *hp*-adaptivity. *Numerical Methods for Partial Differential Equations*, 23(4):832–858, 2007.

[59] Hideko Nakajima, Wei Dong, Elizabeth Olson, Saumil Merchant, Michael Ravicz, and John Rosowski. Differential intracochlear sound pressure measurements

in normal human temporal bones. *JARO - Journal of the Association for Research in Otolaryngology*, 10:23–36, 2009.

[60] Nilima Nigam and Joel Phillips. High-order conforming finite elements on pyramids. *IMA Journal of Numerical Analysis*, 2011.

[61] J.T. Oden, L. Demkowicz, W. Rachowicz, and T.A. Westermann. Toward a universal *hp*-adaptive finite element strategy. Part 2: a posteriori error estimation. *Computer Methods in Applied Mechanics and Engineering*, 77(1-2):113–180, 1989.

[62] J.T. Oden and L.F. Demkowicz. *Applied Functional Analysis.* Chapman and Hall/CRC-Press, 2nd edition, 2010.

[63] J.T. Oden, L.F. Demkowicz, T. Stroubolis, and Ph. Devloo. Adaptive methods for problems in solid and fluid mechanics. In I. Babuška et al., editor, *Accuracy Estimates and Adaptive Refinements in Finite Element Computations*, pages 249–280. John Wiley and Sons, 1986.

[64] Maciej Paszyński, David Pardo, and Anna Paszyńska. Parallel multi-frontal solver for *p*-adaptive finite element modeling of multi-physics computational problems. *Journal of Computational Science*, 1(1):48–54, 2010.

[65] Jörg Peters. Smooth interpolation of a mesh of curves. *Constructive Approximation*, 7:221–246, 1991.

[66] Michael E. Ravicz, Elizabeth S. Olson, and John J. Rosowski. Sound pressure distribution and power flow within the gerbil ear canal from 100 Hz to 80 kHz. *The Journal of the Acoustical Society of America*, 122(4):2154–2173, 2007.

[67] M. Křížek and T. Strouboulis. How to generate local refinements of unstructured tetrahedral meshes satisfying a regularity ball condition. *Numerical Methods for Partial Differential Equations*, 13(2):201–214, 1997.

[68] A. Safjan and J.T. Oden. High-order Taylor-Galerkin methods for linear hyperbolic systems. *Journal of Computational Physics*, 120(2):206–230, 1995.

[69] P. Schiavone, F. Chassat, T. Boudou, E. Promayon, F. Valdivia, and Y. Payan. In vivo measurement of human brain elasticity using a light aspiration device. *Medical Image Analysis*, 13(4):673–678, 2009.

[70] J. Schöberl. Netgen an advancing front 2d/3d-mesh generator based on abstract rules. *Computing and Visualization in Science*, 1(1):41–52, 1997.

[71] Joachim Schöberl and Sabine Zaglmayr. High order Nédélec elements with local complete sequence properties. *COMPEL The International Journal for Computation and Mathematics in Electrical and Electronic Engineering*, 24(2):374–384, 2005.

[72] Will Schroeder, Ken Martin, and Bill Lorensen. The visualization toolkit: an object-oriented approach to 3d graphics. 2006.

[73] Xiquan Shi, Tianjun Wang, Peiru Wu, and Fengshan Liu. Reconstruction of convergent G^1-smooth B-spline surfaces. *Computer Aided Geometric Design*, 21(9):893–913, 2004.

[74] Leon A. Shirman and Carlo H. Séquin. Local surface interpolation with Bezier patches: errata and improvements. *Computer Aided Geometric Design*, 8(3):217–221, 1991.

[75] I. Babuška and W.C. Rheinboldt. Error estimates for adaptive finite element computations. *SIAM Journal on Numerical Analysis*, 15(4):pp. 736–754, 1978.

[76] I. Babuška and T. Strouboulis. *The Finite Element Method and its Reliability.* Oxford University Press, USA, 1st edition, 2001.

[77] P. Solín. *Partial Differential Equations and the Finite Element Method.* Wiley, 2005.

[78] Stefan Stenfelt and Richard Goode. Bone conducted sound: Physiological and clinical aspects. *Otology and Neurotology*, 26(6):1245–1261, 2005.

[79] B.A. Szabó, A. Düster, and E. Rank. The *p*-version of the finite element method. *Encyclopedia of Computational Mechanics*, 1, 2004.

[80] B.A. Szabó and I. Babuška. *Finite Element Analysis.* Wiley, 1991.

[81] R. Tews and W. Rachowicz. Application of an automatic *hp*-adaptive finite element method for thin-walled structures. *Computer Methods in Applied Mechanics and Engineering*, 198(21-26):1967 – 1984, 2009.

[82] Georg v. Bekesy. Vibration of the head in a sound field and its role in hearing by bone conduction. *The Journal of the Acoustical Society of America*, 20(6):749–760, 1948.

[83] R. Verfürth. *A review of a posteriori error estimation and adaptive mesh-refinement techniques*. Wiley-Teubner, 1996.

[84] Dong Xue, Leszek F. Demkowicz, and Chandrajit L. Bajaj. Reconstruction of G^1 surfaces with biquartic patches for *hp*-fe simulations. In *IMR'04*, pages 323–332, 2004.

[85] Mincheol Yoon, Yunjin Lee, Seungyong Lee, Ioannis Ivrissimtzis, and Hans-Peter Seidel. Surface and normal ensembles for surface reconstruction. *Computer-Aided Design*, 39(5):408–420, 2007.

[86] S. Zaglmayr. *High Order Finite Element Methods for Electromagnetic Field Computation*. 2006. Ph.D Thesis, Johannes Kepler University, Linz.

[87] S. Zaglmayr. Higher-order exact sequence for pyramid. 2008. Available from the author.

[88] Yongjie Zhang and Chandrajit Bajaj. Adaptive and quality quadrilateral/hexahedral meshing from volumetric imaging data. In *Computer Methods in Applied Mechanics and Engineering*, 2006.

[89] Yongjie Zhang, Chandrajit Bajaj, and Bong-Soo Sohn. 3d finite element meshing from imaging data. *Computer Methods in Applied Mechanics and Engineering*, 194(48-49):5083–5106, 2005.

Vita

Paolo Gatto was born in Milan, Italy on 23 April 1981, the son of Marino Gatto and Silvia Aglio. He received the Bachelor of Science degree in Applied Mathematics from the University of Pavia in 2003, followed by a Master of Science degree in Mathematics awarded by the same institution in 2006. Both his bachelor and master work were conducted under the supervision of Professor D. Boffi. In August 2007, upon being admitted to the Computational and Applied Mathematics graduate program at the University of Texas at Austin, he moved to Austin, Texas to pursue a Doctor of Philosophy degree under the supervision of Professor L. Demkowicz.

Permanent address: 201 E 24th Street, ACE 4.102
Austin, Texas 78712

This dissertation was typeset with LaTeX[†] by the author.

[†] LaTeX is a document preparation system developed by Leslie Lamport as a special version of Donald Knuth's TeX Program.